

AD-768 649

FINITE ELEMENT ANALYSIS OF STRESSES

DEFORMATIONS AND PROGRESSIVE FAILURE OF NON-
HOMOGENEOUS FISSURED ROCK, VOLUME II, COMPUTER
PROGRAMS USER'S MANUAL

OHIO STATE UNIVERSITY

MARCH 1973

DISTRIBUTED BY:

NTIS

National Technical Information Service
U. S. DEPARTMENT OF COMMERCE

**STRESSES , DEFORMATIONS AND
PROGRESSIVE FAILURE OF
NON-HOMOGENEOUS
FISSURED ROCK**

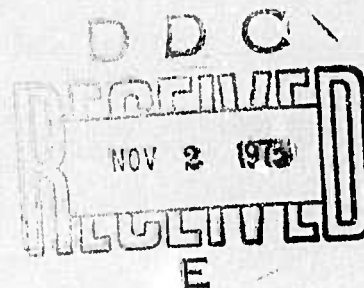
**Final Report
Volume 2 - Computer Program User's Manual
March 1973**

**U.S. BUREAU OF MINES
Contract Number HO210017**

**Sponsored by
ADVANCED RESEARCH PROJECTS AGENCY
ARPA Order No. 1579 , Amend. No. 2
Program Code 1F10**

Principal Investigators

**R. S. Sandhu
T. H. Wu
J. R. Hooper**



The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the Advanced Research Projects Agency or the U.S. Government.



DISTRIBUTION STATEMENT E

**Approved for public release;
Distribution Unlimited**

**THE OHIO STATE UNIVERSITY RESEARCH FOUNDATION
1314 KINNEAR ROAD COLUMBUS, OHIO 43212**

Reproduced by
**NATIONAL TECHNICAL
INFORMATION SERVICE**
U.S. Department of Commerce
Springfield, VA 22151

UNCLASSIFIED

Security Classification

3200.8 (Att 1 to Encl 1)

Mar 7, 66

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author)		2a. REPORT SECURITY CLASSIFICATION	
The Ohio State University Research Foundation		Unclassified	
3. REPORT TITLE		2b. GROUP	
Finite Element Analysis of Stresses, Deformations and Progressive Failure of Non-Homogeneous Fissured Rock - Volume 2, Computer Programs User's Manual			
4. DESCRIPTIVE NOTES (Type of report and inclusive dates)			
Final Report - March 31, 1973			
5. AUTHOR(S) (First name, middle initial, last name)			
Ranbir S. Sandhu			
6. REPORT DATE		7a. TOTAL NO OF PAGES	7b. NO. OF REFS
March 31, 1973		158	None
8a. CONTRACT OR GRANT NO.		9a. ORIGINATOR'S REPORT NUMBER(S)	
HO 210017		OSURF-3177-73-2F	
b. PROJECT NO.		9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)	
RF 3177 A1			
c.			
d.			
10. DISTRIBUTION STATEMENT			
Distribution of this document is unlimited.			
11. SUPPLEMENTARY NOTES		12. SPONSORING MILITARY ACTIVITY	
Computer programs available on Magnetic tape, see AD-768 651.		Advanced Research Projects Agency Washington, D.C. 20301	

13. ABSTRACT The objective of this research program was development of finite element procedures for prediction of stresses and deformations in the vicinity of underground excavation.

Two models of rock behavior were selected. In one the rock is treated as isotropic elastic-plastic following a generalized Mohr-Coulomb law and in the other the rock is isotropic elastic-brittle subject to Griffith or modified Griffith failure theory.

For each model, mathematical relationships governing stress-strain behavior and progressive failure were developed. Finite element computer programs incorporating each of the two models were coded. Preliminary to this development, a revised version of Zienkiewicz's no-tension analysis was programmed.

The procedures developed allow for initial stresses in rock, arbitrary shape and size of the opening, any given sequence of construction/excavation, material nonhomogeneity and progressive failure.

This report is in three parts: Volume 1-Main Document; Volume 2-Computer Program User's Manual; Volume 3-Computer Programs.

Volume 2 -Computer Program User's Manual, contains descriptions, instructions for usage and fortran listings of computer programs used to obtain the numerical results presented and discussed in Volume 1 -Main Document.

DD FORM 1 NOV 55 1473

UNCLASSIFIED

Security Classification

Unclassified

Security Classification

3200.8 (Att 1 to Encl 1)

Mar 7, 66

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
computation crack propagation deformation elasticity excavation failure finite element method foundation fracture mining plasticity progressive failure research rock mechanics stresses structural supports tunnels underground excavation						

Unclassified

Security Classification

1b

FINAL REPORT

ARPA Order Number: 1579, Amend 2

Contract Number: HO210017

Program Code Number: 1F10

Principal Investigators: R. S. Sandhu
T. H. Wu
J. R. Hooper

Telephone Number: (614) 422-7531

Name of Contractor:
The Ohio State University
Research Foundation

Project Scientist or Engineer:
R. S. Sandhu
Telephone Number: (614) 422-7531

Effective Date of Contract:
February 1, 1971

Short Title of Work:
Stresses, Deformations and
Progressive Failure of
Nonhomogeneous Fissured Rock

Contract Expiration Date:
March 31, 1973

Amount of Contract:
\$71,613.00

This research was supported by the Advanced Research Projects Agency of the Department of Defense and was monitored by the United States Bureau of Mines under Contract Number HO210017.

Distribution of this document is unlimited.

10

FOREWORD

The final report on work done under contract HO 210017 between the Ohio State University and the United States Bureau of Mines is in three parts as follows:

- | | |
|-----------|--------------------------------|
| Volume 1: | Main Document |
| Volume 2: | Computer Program User's Manual |
| Volume 3: | Computer Programs |

Volume 2 of the report contains documentation relating to three computer programs including fortran listings and description of program structure.

It is obvious that these computer programs be used only under the conditions and assumptions for which they were developed. These are described in Volume 1 of this report. Although the programs have been tested by applications to several problems, no warranty is made regarding the accuracy and reliability of the programs and no responsibility is assumed by the authors or by the sponsors of this research project.

The technical report summary is included in all three volumes of the report.

R. S. Sandhu
Principal Investigator

TECHNICAL REPORT SUMMARY

Program Objectives

The objective of this research program was development of finite element procedures to predict stresses, deformations and progressive failure of rock associated with underground excavations. For applicability to arbitrary sequence of excavation operations, it was necessary that the procedures developed allow for arbitrary initial stresses in rock, arbitrary size and shape of the opening and progressive failure. Plane strain conditions and two different types of material behavior were considered. Rock was treated as an isotropic elastic-plastic generalized Mohr-Coulomb material in one model and as an elastic-brittle material following Griffith theory of fracture in the other.

Background

In previous applications of the finite element method to rock mechanics, elastic-plastic behavior of rock has been modeled as nonlinear elastic for computational convenience. Further, it was assumed that the results of a one-dimensional test could be generalized to three-dimensional analysis through the use of an equivalent stress-equivalent strain curve. In some applications, two stress or strain parameters were used. These procedures are unsatisfactory. Assumption of isotropic elasticity assumes that the principal directions of stress and strain coincide. In plasticity this is not true. Also, rock behavior is characterized by a significant part of deformation being irreversible. For this reason, the mechanical behavior in unloading is different from that in loading. For rock with preexisting joints or developing tensile cracks, a 'no tension'

procedure is often adopted. In this method, a linear elastic solution is obtained and all tensile stress redistributed simultaneously. Actually, as cracking progresses, the rock on either side of the crack is relieved of stress and a stress concentration develops near the crack tip. Conventional procedures ignore these effects and the progressive nature of crack development, leading to erroneous conclusions regarding stresses around underground openings.

Accomplishments Under the Present Program

The research conducted under this contract has resulted in development of computer programs based on more realistic simulation of material behavior. The incremental theory of plasticity has been used to characterize the stress-strain behavior of elastic-plastic rock. Role of kinematic constraint of plane strain in development of residual stresses in rock has been examined on the basis of Hill's theory. New techniques have been developed for study of initiation and propagation of fracture in rock following Griffith's theory or the modified Griffith theory. Allowing for sequential fracture of various elements in a system, the effect of progressive stress redistribution in the remaining system is correctly incorporated. Arbitrary initial stress states, arbitrary sequence of excavation (or construction), arbitrary size and shape of opening, and nonhomogeneous material properties were allowed for. The actual construction operations can be simulated. The procedures developed were applied to several typical problems in rock mechanics as well as to some theoretical and laboratory studies for the purpose of verification and illustration. These were used to carry out parametric studies to examine the influence of rock properties upon the stresses in steel supports in a tunnel.

DOD Implications

The procedures developed provide useful means for study of stability of underground excavations based on stresses and deformations associated with the mining operations, structural support evaluation, safety analyses of openings, study of blasting effectiveness under certain conditions, evaluation of mining sequences, study of vulnerability and serviceability of underground structures etc.

Organization of the Report

This report is in three parts as follows:

- Volume 1 - Main Document
- Volume 2 - Computer Program User's Manual
- Volume 3 - Computer Programs

Volume 1 contains the main body of the report including the theoretical development, program verification and case studies. Chapter I reviews previous efforts in the general research area and describes the objectives and methods of the present research in the historical context. Chapter II describes the mechanical behavior of rock and the idealizations used in the research under report. The basis and methods of the finite element theory are briefly discussed in Chapter III leading to the formulation of matrix equations. Chapter IV gives details of the analysis technique for isotropic elastic-plastic generalized Mohr-Coulomb rock materials and Chapter V gives the numerical analysis procedure for jointed rock and rock subjected to progressive fracture following Griffith or modified Griffith theory. Examples of application are included in Chapters IV and V. Chapter VI presents application of the elastic-plastic analysis computer program to a parametric study to evaluate the influence of rock properties on stresses in steel supports for specified initial stresses and design of the opening.

In the original proposal, model testing to verify some aspects of rock behavior under plane strain conditions was foreseen. The effort under the present contract covered procurement of suitable plane strain test equipment and design of suitable test material. Appendix B includes a report on this effort.

Volume 2 of the report contains description of the three computer programs developed under the contract along with fortran listings and instructions for input preparation. The input definition and the listings are for the IBM 370/165 version.

The programs are the primary content of volume 3. These are available on magnetic tape from DDC-TC, U.S. Department of Commerce, Springfield, Virginia 22151, telephone (703) 321-8517.

ACKNOWLEDGEMENTS

The research was supported by the U.S. Government through the Advanced Research Projects Agency, ARPA, and its agent the U.S. Bureau of Mines, Department of the Interior. James J. Olson, Twin Cities Mining Research Center, was the ARPA program coordinator and Dr. William J. Karwoski, Spokane Mining Research Center, was the Project Officer. In early stages of work, Dr. Syd Peng, Twin Cities Mining Research Center acted as the Project Officer. Constant cooperation and several constructive suggestions from these individuals are gratefully appreciated.

A number of graduate students worked on the project. The contributions of Messrs. Ram Dhan Singh, S. W. Huang and Kamar Jit Singh were specially noteworthy. The Instruction and Research Computer Center of the Ohio State University provided the computational facilities.

R. S. Sandhu
Project Supervisor

TABLE OF CONTENTS

	<u>Page</u>
FOREWORD	i
TECHNICAL REPORT SUMMARY	ii
ACKNOWLEDGEMENTS	vi
TABLE OF CONTENTS	vii
CHAPTER I: NOTENS--A Computer Program for Two-Dimensional Analysis of Rock as a No Tension Material	1
1.1. Purpose and Capability	1
1.2. Program Organization	1
1.3. Input Data	4
1.4. Output Information	8
1.5. Limitations	8
1.6. Fortran Listing	9
CHAPTER II: ELPL--A Computer Program for Plane Strain Analysis of Progressive Failure of Nonhomogeneous Rock Following Incremental Theory of Plasticity, Plane Strain Conditions, Generalized Mohr-Coulomb Yield Criterion	30
2.1. Purpose and Capability	30
2.2. Program Organization	30
2.3. Input Data	36
2.4. Output Information	45
2.5. Fortran Listing	46
CHAPTER III: GRIFTH--A Computer Program for Two-Dimensional Analysis of Progressive Failure of Rock Following Griffith and Modified Griffith Theory	76
3.1. Purpose and Capability	76
3.2. Organization	76
3.3. Input Data	78
3.4. Output Information	86
3.5. Fortran Listing	87

CHAPTER I: NOTENS-- A Computer Program for Analysis of Rock as a No-Tension Material

1.1. Purpose and Capability

This computer program is based on a modification of Zienkiewicz's approach (Zienkiewicz et.al. 1968). A linear elastic analysis of a given two-dimensional system is carried out to determine the regions of tensile stress. The tensions, wherever they occur, are replaced by equivalent nodal point loads and the tensile stresses in the corresponding elements are neutralized. For these equivalent loads, the elastic analysis is repeated. This iterative process is continued until all tensile stresses are liquidated. The modification to the original approach is discussed in Part I: Technical Report.

The program allows for an arbitrary initial stress field, gravity loads, concentrated loads, distributed loads applied to the boundary, temperature loads and temperature dependent material properties.

1.2. Organization

The program is in Fortran language. Scratch tapes are 1 and 2. Tapes 5 and 6 are used for input/output respectively. The listing in section 1.6 used double precision for real numbers. The program capacity can be altered by changing the dimension of arrays AA and IA. These correspond to the total locations required in real arrays and integer arrays respectively. NTOT, MTOT at lines MAIN 23, MAIN 24 are set equal to the dimension of AA, IA respectively.

The program consists of the following units:

a. Program MAIN

In this unit, the control information is read in and the location of various dimensioned variables defined in the system arrays AA and IA.

b. Subroutine INPT

This subroutine is called by the MAIN. In this, information regarding material properties; nodal point coordinates, loads, temperatures, boundary conditions; element geometry, material and initial stresses is read in. Missing information is generated. Information regarding boundary pressure is read in. Dimensions of the matrix for solution of stiffness equations are defined and subroutine SOLVE is called to complete the solution process.

c. Subroutine SOLVE

Called by INPT after reading in all data, this subroutine sets up the system stiffness matrix in blocks. It calls subroutines ONED and ELEMEN to obtain stiffness properties for one-dimensional and two-dimensional elements respectively. The stiffness is assembled by the direct stiffness procedure and modified for specified boundary displacements. The matrix is triangularized by calling SYMBAN (1). Subroutine LOAD is called to recover the system load vector. Reduction of the load vector and back-substitution are accomplished by calling SYMBAN (2). The cumulative as well as the incremental displacements are pointed out. Subroutine STRESS is called for evaluation and print out of stresses. Because the solution is obtained by elimination of tension in an iterative sequence, the steps from calling LOAD to calculation of stress are repeated a preset number of times or until convergence is obtained.

d. Subroutine ONED

Called by SOLVE, this subroutine sets up the stiffness matrix for a one-dimensional element.

e. Subroutine ELEMEN

Called by SOLVE, this subroutine calculates the stiffness matrix for two-dimensional elements. These may be triangular elements or quadrilaterals made up of four constant strain triangles. If other types of quadrilateral elements are to be used, this subroutine has to be replaced. Linear isotropic elasticity is assumed. Loads due to initial stresses or temperature changes from a reference temperature, and gravity loads are calculated in this subroutine.

f. Subroutine MODIFY

Called by SOLVE, this subroutine modifies the system stiffness matrix to allow for prescribed displacement boundary conditions.

g. Subroutine SYMBAN

This subroutine is called by SOLVE. Called with augment 1, it triangularizes the stiffness matrix and called with augment 2, it backsubstitutes to evaluate the solution for displacements for a given load vector.

h. Subroutine LOAD

This subroutine called by SOLVE, sets up the system load vector including nodal point loads, distributed boundary pressure loads, as well as unbalanced element loads, initial stresses, temperature stresses and gravity loads.

j. Subroutine STRESS

Stresses in x-y coordinate system as well as principal stresses are calculated in the STRESS subroutine called by SOLVE. The stresses are checked for tension, and if necessary, the tensions are replaced by equivalent unbalanced stresses

to be transformed into load in the next cycle. A check on convergence of the iterative scheme is applied and the maximum unbalanced stress in any element is printed out.

1.3. Input Data

Input to the program NOTENS consists of the following sequence of cards.

a. First Card. Job Title (18A4).

This card gives the descriptive identification of the job.

b. Second Card. Control Information (4I5, 3F10.2, I5, D15.4)

<u>Information</u>	<u>Columns</u>
Total number of nodal points (NUMNP)	1-5
Total number of elements (NUMEL)	6-10
Number of different materials (NUMMAT)	11-15
Number of pressure boundary cards (NUMPL)	16-20
Acceleration in x-direction (ACELR)	21-30
Acceleration in y-direction (ACELZ)	31-40
Reference (stress-free) temperature (Q)	41-50
Maximum number of approximations or iterations allowed (NP)	51-55
Tolerance for convergence of iterations (TOL)	56-70

c. Material Property Cards

One set of cards must be provided for each material. In each set:

i. First card (2I5, 1F10.3, I5) gives the following information:

Material identification number ($1 \leq \text{MTYPE} \leq 10$)	1-5
Number of temperature cards (NTC)	6-10

Mass density of the material (RO) 11-20

Material code (MTC): 1 for no-tension material 21-25
0 otherwise

ii. Subsequent cards, one for each temperature, the number of data sets defined in columns 6-10 of the first card for the material, will carry the the following information (4F10.3):

<u>Information</u>	<u>Columns</u>
Temperature	1-10
Elastic Modulus	11-20
Poisson's ratio	21-30
Coefficient of thermal expansion (or cross-sectional area of 1-D element)	31-40

d. Nodal Point Cards (I5, F5.1, 5 F10.4)

Nodal point number (N)	1-5
Code of nodal point (CODE)	6-10
X-ordinate (R)	11-20
Y-ordinate (Z)	21-30
UR	31-40
UZ	41-50
Temperature (T)	51-60

If the number in columns 6-10 is

0. UR is the specified X-load and UZ is the specified Y-load
1. UR is the specified X-displacement and UZ is the specified Y-load
2. UR is the specified X-load and UZ is the specified Y-displacement
3. UR is the specified X-displacement and UZ is the specified Y-displacement

All loads are considered to be total forces acting on an element of unit thickness. Nodal point cards must be in numerical sequence. If cards are omitted, the omitted nodal points are generated at equal intervals along a straight line between the defined nodal points. Similarly, the corresponding temperatures are determined by linear interpolation. The codes (CODE) of these generated nodal points, as well as UR and UZ, are set equal to 0.

e. Element cards

i. Material type information of elements (16I5)

Every 5 columns of each data card give the material identification number of each element in sequence. Each card contains 16 material identification numbers (80 columns of 16 elements).

ii. Nodal points and initial stresses (5I5, 5X, 3F10.0). One card for each element.

<u>Information</u>	<u>Columns</u>
Number of the element	1-5
Nodal point I	6-10
Nodal point J	11-15
Nodal point K	16-20
Nodal point L	21-25
Initial σ_{xx}	31-40
Initial σ_{yy}	41-50
Initial σ_{xy}	51-60

Nodal points I, J, K, L are corners of each individual element in a counter-clockwise order for a right-handed system of co-ordinates. For 1-D elements,

nodal points K and L are set same as J and I respectively. In the case of triangular elements, L is set to be same as K. The element cards must be in numerical sequence. Any element cards omitted will be automatically generated in the program by incrementing each of the I, J, K and L nodal points by 1. The corresponding initial stresses of the generated elements are calculated by the program using linear interpolation. Such interpolation is done basing upon element number rather than distance.

f. Pressure Boundary cards (2I5, 2F10.3)

One card for each boundary element, which is subjected to a normal pressure, will carry the following information:

<u>Information</u>	<u>Columns</u>
Nodal point I (IBC)	1-5
Nodal point J (JBC)	6-10
Normal pressure at I (PR)	11-20
Normal pressure at J (PR)	21-30



As shown in the sketch, the boundary element must be on the left as one progresses counter-clockwise from I to J. Surface tensile force is input as a negative pressure.

1.4. Output Data

The following information is developed and printed by the program:

- a. Printing of input (and generated)data
- b. Incremental and cumulative nodal displacements of each iteration
- c. Stresses at center of each element and the unbalanced force of the elements during each iteration
- d. Maximum unbalanced force of the elements for the iteration
- e. In the case where maximum unbalanced force is less than the set tolerance (TOL), convergence is noted and the number of cycles to reach this is printed.

1.5. Limitations

Limitations of the approach are discussed in detail in Part I-Technical Report. In this approach it is assumed that all the elements having tensile stress crack and are relieved of tension simultaneously. This is unrealistic. Actually the process of cracking will be sequential with cracking of each element influencing the stress distribution and consequently the continuation of the cracking sequence. The program included here eliminates some of the procedural errors of earlier attempts. However, it cannot predict sequential or progressive development of cracks. Also convergence is often very slow and the advantage of using the same system stiffness matrix throughout the iterative computation is lost because of the large number of iterations required.

1.6. Fortran Listing

```

MAIN 1 C *****
MAIN 2 C PROGRAM NOTENS
MAIN 3 C NO TENSION ANALYSIS FOR PLANE STRESS AND PLANE STRAIN.
MAIN 4 C LINEAR PRESSURE BOUNDARY
MAIN 5 C PROGRAMMED BY R.S.SANDHU, R.D.SINGH AND H.LIU. THE OHIO STATE
MAIN 6 C UNIVERSITY, COLUMBUS.
MAIN 7 C THE FORMULATION IS DOCUMENTED IN THE FINAL REPORT DATED MARCH 31,
MAIN 8 C 1973, ON CONTRACT H0210017 BETWEEN THE OHIO STATE UNIVERSITY AND
MAIN 9 C THE UNITED STATES BUREAU OF MINES SUPPORTED BY THE ADVANCED
MAIN 10 C RESEARCH PROJECTS AGENCY. INSTRUCTIONS FOR USE OF THE PROGRAM
MAIN 11 C ARE CONTAINED IN PART II OF THE REPORT.
MAIN 12 C *****
MAIN 13 C
MAIN 14 C IMPLICIT REAL*8(A-H,O-Z)
MAIN 15 C COMMON AA(10000),IA(1500)
MAIN 16 C COMMON/ONE/ACFLK,ACELZ,TEMP,W,TOL,VOL,NUMNP,NUMEL,NUMMAT,NUMPC,
MAIN 17 C *MYPE,LLL,N,MBAND,NUMBLK,KKK,NCHECK,NP,NL,NFQ,HED(18),NPC
MAIN 18 C COMMON/TWO/C(3,3),S(10,10),SIG(6),P(6),ST(3,10),RR(5),ZZ(5),XC,YC,
MAIN 19 C *EE(3),LM(4),E(5,4,10),RC(10),NTC(10),MTC(10)
MAIN 20 C COMMON/THREE/MTOT,NTCT,N11,M4,STOP
MAIN 21 C READ(5,1000) HED,NUMNP,NUMEL,NUMMAT,NUMPC,ACFLK,ACELZ,Q,NP,TOL
MAIN 22 C WRITE(6,2000) HED,NUMNP,NUMEL,NUMMAT,NUMPC,ACFLK,ACELZ,Q,NP,TOL
MAIN 23 C N10T=10000
MAIN 24 C MTCT=1500
MAIN 25 C NPC=NUMPC
MAIN 26 C IF (NUMPC.EQ.0) NPC=1
MAIN 27 C N1=1
MAIN 28 C N2=N1+NUMNP
MAIN 29 C N3=N2+NUMNP
MAIN 30 C N4=N3+NUMNP
MAIN 31 C N5=N4+NUMNP
MAIN 32 C N6=N5+NUMNP
MAIN 33 C N7=N6+2*NPC
MAIN 34 C N8=N7+6*NUMEL
MAIN 35 C N9=N8+NUMNP
MAIN 36 C N10=N9+2*NUMNP
MAIN 37 C N11=N10+2*NUMNP
MAIN 38 C M1=1
MAIN 39 C M2=M1+5*NUMEL
MAIN 40 C M3=M2+NPC
MAIN 41 C M4=M3+NPC
MAIN 42 C NLQ=2*NUMNP
MAIN 43 C JJ=M4-MTCT
MAIN 44 C IF (JJ.LE.0) GO TO 100
MAIN 45 C WRITE(6,3000) JJ
MAIN 46 C CALL EXIT
MAIN 47 C 100 CONTINUE
MAIN 48 C CALL INPT(AA(N1),AA(N2),AA(N3),AA(N4),AA(N5),AA(N6),AA(N7),
MAIN 49 C *AA(N8),AA(N9),AA(N10),IA(M1),IA(M2),IA(M3))
MAIN 50 C 1000 FORMAT(18A4/4I5,3F10.2,15,D15.4)

```

```

MAIN 51 2000 FORMAT (1H1 18A4/
MAIN 52 1 30H0 NUMBER OF NODAL POINTS----- 13 /
MAIN 53 2 30H0 NUMBER OF ELEMENTS----- 13 /
MAIN 54 3 30H0 NUMBER OF DIFF. MATERIALS--- 13 /
MAIN 55 4 30H0 NUMBER OF PRESSURE CARDS---- 13 /
MAIN 56 5 30H0 X-ACCELERATION----- E12.4/
MAIN 57 6 30H0 Y-ACCELERATION----- E12.4/
MAIN 58 7 30H0 REFERENCE TEMPERATURE----- E12.4/
MAIN 59 8 30H0 NO. OF APPROXIMATIONS----- 15/
MAIN 60 9 30H0 TOLERANCE FOR CONVERGENCE--- E12.4)
MAIN 61 3000 FORMAT (70H PROGRAM EXECUTION TERMINATED. REQUIRED CORE EXCEEDS NT
MAIN 62 *OT BY 110)
MAIN 63 END

```

```

INPT 1      SUBROUTINE INPT(R,Z,UR,UZ,T,PR,SIG1,CODE,b,CU,IX,IBC,JBC)
INPT 2      IMPLICIT REAL*8(A-H,G-Z)
INPT 3      COMMON AA(10000),IA(1500)
INPT 4      COMMON/ONE/ACFLK,ACFL2,TEMP,Q,TUL,VOL,NUMNP,NUMEL,NUMMAT,NUMPC,
INPT 5      *MTYPE,LLI,N,M6AND,NUMBLK,KKK,NCHECK,NP,NL,NFQ,HEQ(18),NPC
INPT 6      COMMON/TWO/C(3,3),S(10,10),SIG(6),P(6),ST(3,10),RR(5),ZZ(5),XC,YC,
INPT 7      *FF(3),LM(4),E(5,4,10),RU(10),NTC(10),MTC(10)
INPT 8      COMMON/THREE/MTOT,NTOT,N11,M4,SIDP
INPT 9      DIMENSION R(NUMNP),Z(NUMNP),UR(NUMNP),UZ(NUMNP),T(NUMNP),PP(NPC,2)
INPT 10     *,SIG1(NUMEL,6),CODE(NUMNP),IX(NUMEL,5),IBC(NPC),JBC(NPC)
INPT 11     DIMENSION B(NFQ),CU(NEL)
INPT 12     DO 50 M=1,NUMMAT
INPT 13     READ (5,1001) MTYPE,NTC(MTYPE),RU(MTYPE),MTC(MTYPE)
INPT 14     WRITE(6,2001) MTYPE,NTC(MTYPE),RU(MTYPE),MTC(MTYPE)
INPT 15     NUMTC=NTC(MTYPE)
INPT 16     READ (5,1002) ((I(1,J,MTYPE),J=1,4),I=1,NUMTC)
INPT 17     WRITE (6,2002) ((F(1,J,MTYPE),J=1,4),I=1,NUMTC)
INPT 18     50 CONTINUE
INPT 19     WRITE (6,2003)
INPT 20     L=0
INPT 21     60 READ (5,1003) N,CODE(N),R(N),Z(N),UR(N),UZ(N),T(N)
INPT 22     NL=L+1
INPT 23     IF(N.EQ.1) GO TO 70
INPT 24     ZX=N-L
INPT 25     DR=(R(N)-R(L))/ZX
INPT 26     DZ=(Z(N)-Z(L))/ZX
INPT 27     DT=(T(N)-T(L))/ZX
INPT 28     70 L=L+1
INPT 29     IF(N-L) 100,90,80
INPT 30     80 CODE(L)=0.0
INPT 31     R(L)=R(L-1)+DR
INPT 32     Z(L)=Z(L-1)+DZ
INPT 33     T(L)=T(L-1)+DT
INPT 34     UR(L)=0.0
INPT 35     UZ(L)=0.0
INPT 36     GO TO 70
INPT 37     90 CONTINUE
INPT 38     IF(NUMNP=N) 100,110,60
INPT 39     100 WRITE (6,2005) N
INPT 40     CALL EXIT
INPT 41     110 WRITE (6,2004) ((K,CODE(K),R(K),Z(K),UR(K),UZ(K),T(K)),K=1,NUMNP)
INPT 42     READ(5,1007) (IX(N,5),N=1,NUMEL)
INPT 43     WRITE (6,2006)
INPT 44     N=0
INPT 45     130 READ (5,1004) M,(IX(M,1),I=1,4) ,(SIG1(M,1),I=1,3)
INPT 46     IF(M.EQ.1) GO TO 140
INPT 47     ZX=M-N
INPT 48     DO 135 I=1,3
INPT 49     135 SIG(I)=(SIG1(M,1)-SIG1(N,1))/ZX
INPT 50     140 N=N+1

```



```

INPT 51      IF(M.LE.N)GO TO 170
INPT 52      IX(N,1)=IX(N-1,1)+1
INPT 53      IX(N,2)=IX(N-1,2)+1
INPT 54      IX(N,3)=IX(N-1,3)+1
INPT 55      IX(N,4)=IX(N-1,4)+1
INPT 56      DO 160 I=1,3
INPT 57      160 SIGI(N,I)=SIGI(N-1,I)+SIG(I)
INPT 58      170 IF(M.GT.N) GO TO 14C
INPT 59      IF(N.LT.NUMEL)GO TO130
INPT 60      WRITE(6,2007)((N,(IX(N,I),I=1,5),(SIGI(N,I),I=1,3)),N=1,NUMEL)
INPT 61      IF(NUMPC.EQ.0) GO TO 310
INPT 62      290 WRITE (6,2008)
INPT 63      DO 300 L=1,NUMPC
INPT 64      READ(5,1005) IRC(L),JBC(L),PR(L,1),PR(L,2)
INPT 65      300 WRITE(6,2009) IRC(L),JBC(L),PR(L,1),PR(L,2)
INPT 66      310 CONTINUE
INPT 67      J=0
INPT 68      DO 340 N=1,NUMEL
INPT 69      DO 340 I=1,4
INPT 70      DO 325 L=1,4
INPT 71      KK=IABS(IX(N,I)-IX(N,L))
INPT 72      IF(KK.GT.J) J=KK
INPT 73      325 CONTINUE
INPT 74      340 CONTINUE
INPT 75      MBAND=2*J+2
INPT 76      WRITE(6,3000)MBAND
INPT 77      NL=(NTOT-N11+1)/MBAND
INPT 78      NLL=NEQ+3
INPT 79      IF(NL.GT.NLL) NL=NLL
INPT 80      NL=NL/4
INPT 81      NL=4*NL
INPT 82      NBAND=2*MBAND
INPT 83      IF(NL.GE.NBAND)GO TO 350
INPT 84      WRITE (6,4010)NL,MBAND
INPT 85      CALL EXIT
INPT 86      350 CONTINUE
INPT 87      N12=N11+NL*MBAND
INPT 88      JJ=N12-NTOT
INPT 89      IF(JJ.LE.0)GO TO 360
INPT 90      WRITE(6,3050)JJ
INPT 91      CALL EXIT
INPT 92      360 CONTINUE
INPT 93      WRITE(6,4000)N12,M4
INPT 94      CALL SOLVE(R,Z,UR,UZ,T,PR,SIGI,CODE,B,CU,AA(N11),IX,
INPT 95      *IRC,JBC)
INPT 96      RETURN
INPT 97      1001 FORMAT (215,1F10.3,15)
INPT 98      1002 FORMAT (4F10.5)
INPT 99      1003 FORMAT (15,F5.1,5F10.4)
INPT100      1004 FORMAT(515,5X,3F10.0)

```

```

INPT101 1005 FORMAT(2F10.3)
INPT102 1007 FORMAT(16I5)
INPT103 2001 FORMAT (17HOMATERIAL NUMBER= 13, 30H, NUMBER OF TEMPERATURE CARDS-
INPT104 1 13, 15H, MASS DENSITY= F12.4,16H, MATERIAL CODE= 15)
INPT105 2002 FORMAT (15H0 TEMPERATURE 10X 5HE 9X 6HNU 10X 5HALPHA/
INPT106 1(F15.2,3F15.5))
INPT107 2003 FORMAT (10BHINODAL POINT TYPE X ORDINATE Y ORDINATE X LC
INPT108 1AD OR DISPLACEMENT Y LOAD OR DISPLACEMENT TEMPERATURE )
INPT109 2004 FORMAT (112,F12.2,2F12.5,2E24.7,F12.3)
INPT110 2005 FORMAT (26HINODAL POINT CARD ERROR N= 15)
INPT111 2006 FORMAT(10BH1ELEMENT NO. I J K L MATERIAL )
INPT112 1SIGIXX SIGIYY SIGIXY
INPT113 2007 FORMAT (1113,4I6,1112,3F11.3)
INPT114 2008 FORMAT(29HOPRESSURE BOUNDARY CONDITIONS/40H I J PRESSURE
INPT115 1E I PRESSURE J)
INPT116 2009 FORMAT(216,2F14.5)
INPT117 3000 FORMAT(30H BAND WIDTH----- 15)
INPT118 3050 FORMAT (70H PROGRAM EXECUTION TERMINATED. REQUIRED CORE EXCEEDS NT 110)
INPT119 *OT 6Y 15,
INPT120 4000 FORMAT (47H FOR THIS PROGRAM THE LOCATION USED IN AA IS = 15,
INPT121 *17H AND IN IA IS = 15)
INPT122 4010 FORMAT (25H0 NL IS LESS THAN 2*MBAND /
INPT123 *5H0 NL= 15/
INPT124 *8H0 MBAND= 15)
INPT125 END

```

```

SOLV 1      SUBROUTINE SOLVE(R,Z,UR,UZ,T,PR,SIG1,CODE,B,CU,A,IX,IBC,JBC)
SOLV 2      IMPLICIT REAL*8(A-H,O-Z)
SOLV 3      COMMON AA(10000),IA(1500)
SOLV 4      COMMON/ONE/ACELR,ACELZ,TEMP,Q,TOL,VOL,NUMNP,NUMEL,NUMMAT,NUMPC,
SOLV 5      *MYPE,LLL,N,MBAND,NUMBLK,KKK,NCHECK,NP,NL,NFQ,HED(18),NPC
SOLV 6      COMMON/TWO/C(3,3),S(10,10),SIG(6),P(8),ST(3,10),RR(5),ZZ(5),XC,YC,
SOLV 7      *EE(3),LM(4),E(5,4,10),RU(10),NTC(10),MTC(10)
SOLV 8      COMMON/THREE/MTOT,NTOT,N11,M4,STUP
SOLV 9      DIMENSION R(NUMNP),Z(NUMNP),UR(NUMNP),UZ(NUMNP),T(NUMNP),PR(NPC,2)
SOLV 10     *,SIG1(NUMEL,6),CODE(NUMNP),IX(NUMEL,5),IBC(NPC),JBC(NPC)
SOLV 11     DIMENSION CU(NEQ),A(NL,MBAND),E(NEQ)
SOLV 12     DO 10 N=1,NUMEL
SOLV 13     DO 10 I=4,6
SOLV 14     10 SIG1(N,I)=0.
SOLV 15     DO 20 N=1,NUMNP
SOLV 16     NN=2*N
SOLV 17     CU(NN-1)=0.
SOLV 18     20 CU(NN)=0.
SOLV 19     REWINO 2
SOLV 20     ND2=NL
SOLV 21     NO=NL/2
SOLV 22     NB=NO/2
SOLV 23     STUP=0.0
SOLV 24     NUMBLK=0
SOLV 25     DO 50 N=1,ND2
SOLV 26     DO 50 M=1,MBAND
SOLV 27     50 A(N,M)=0.0
SOLV 28     60 NUMBLK=NUMBLK+1
SOLV 29     NH=NB*(NUMBLK+1)
SOLV 30     NM=NH-NB
SOLV 31     NZ=NM-NB+1
SOLV 32     KSHIFT=2*NZ-2
SOLV 33     DO 210 N=1,NUMEL
SOLV 34     NN=IX(N,1)
SOLV 35     DO 65 I=2,4
SOLV 36     IF(IX(N,I).LT.NN) NN=IX(N,I)
SOLV 37     65 CONTINUE
SOLV 38     IF((NN.LT.NZ).OR.(NN.GT.NM)) GO TO 210
SOLV 39     IF(IX(N,3).NE.IX(N,2)) GO TO 95
SOLV 40     CALL ONED(R,Z,UR,UZ,T,PR,SIG1,CODE,CU,A,B,IX,IBC,JBC)
SOLV 41     MM=2
SOLV 42     GO TO 130
SOLV 43     95 CALL ELEMEN(R,Z,UR,UZ,T,PR,SIG1,CODE,CU,A,B,IX,IBC,JBC)
SOLV 44     IF(VOL.GT.0)GO TO 110
SOLV 45     100 WRITE(6,2000) N
SOLV 46     STOP=1.0
SOLV 47     110 MM=4
SOLV 48     IF(IX(N,3).EQ.IX(N,4))MM=3
SOLV 49     130 CONTINUE
SOLV 50     DO 140 I=1,MM

```

```

SOLV 51 140 LM(1)=2*IX(N,1)-2
SOLV 52 DO 200 I=1,MM
SOLV 53 DO 200 K=1,2
SOLV 54 II=LM(1)+K-KSHIFT
SOLV 55 KK=2*I-2+K
SOLV 56 DO 200 J=1,MM
SOLV 57 DO 200 L=1,2
SOLV 58 JJ=LM(J)+L-II+1-KSHIFT
SOLV 59 LL=2*J-2+L
SOLV 60 IF(JJ.LE.0) GO TO 200
SOLV 61 IF(ND.GE.JJ) GO TO 195
SOLV 62 180 WRITE (6,2001) N
SOLV 63 STOP=1.0
SOLV 64 GO TO 210
SOLV 65 195 A(II,JJ)=A(II,JJ)+S(KK,LL)
SOLV 66 200 CONTINUE
SOLV 67 210 CONTINUE
SOLV 68 DO 400 M=NZ,NH
SOLV 69 IF(M.GT.NUMNP) GO TO 410
SOLV 70 N=2*M-1-KSHIFT
SOLV 71 IF(CODE(M)) 340,400,316
SOLV 72 316 IF(CODE(M).EQ.1.)GO TO 370
SOLV 73 IF(CODE(M).EQ.2.)GO TO 340
SOLV 74 GO TO 380
SOLV 75 370 CALL MODIFY(A,ND2,MBAND,N)
SOLV 76 GO TO 400
SOLV 77 380 CALL MODIFY(A,ND2,MBAND,N)
SOLV 78 340 CONTINUE
SOLV 79 N=N+1
SOLV 80 CALL MODIFY(A,ND2,MBAND,N)
SOLV 81 400 CONTINUE
SOLV 82 410 CONTINUE
SOLV 83 WRITE(2) ((A(N,M),M=1,MBAND),N=1,ND)
SOLV 84 DO 420 N=1,ND
SOLV 85 K=N+ND
SOLV 86 DO 420 M=1,MBAND
SOLV 87 A(N,M)=A(K,M)
SOLV 88 420 A(K,M)=0.0
SOLV 89 IF(NM.LT.NUMNP) GO TO 60
SOLV 90 IF(STOP.NE.0.)CALL EXIT
SOLV 91 NCHECK=1
SOLV 92 KKK=1
SOLV 93 CALL SYMBAN(ND2,A,B,NFQ,MBAND,NUMBLK,KKK)
SOLV 94 KKK=2
SOLV 95 DO 550 LLL=1,NP
SOLV 96 CALL LOAD(K,Z,UR,UZ,T,PR,SIGI,CODE,CU,A,B,IX,IEC,JRC)
SOLV 97 CALL SYMBAN(ND2,A,B,NEQ,MBAND,NUMBLK,KKK)
SOLV 98 DO 510 N=1,NUMNP
SOLV 99 NN=2*N
SOLV100 CU(NN-1)=CU(NN-1)+B(NN-1)

```



```

SOLV101 510 CU(NN)=CU(NN)+B(NN)
SOLV102 WRITE (6,2013) LLL
SOLV103 WRITE(6,2010) (N,B(2*N-1),B(2*N),CU(2*N-1),CU(2*N), N = 1,NUMNP)
SOLV104 CALL STRESS(R,Z,UR,UZ,T,PR,SIG1,CODE,CU,A,B,IX,IBC,J6C)
SOLV105 IF(NCHECK.EQ.0) GO TO 600
SOLV106 550 CONTINUE
SOLV107 GO TO 990
SOLV108 600 WRITE (6,2011)LLL
SOLV109 990 RETURN
SOLV110 2000 FORMAT (26HNEGATIVE AREA ELEMENT NO. 14)
SOLV111 2001 FORMAT (29HOBAND WIDTH EXCEEDS ALLOWABLE 14)
SOLV112 2010 FORMAT(12HON.P.NUMBER 17X 3HDUX 17X 3HDUY 18X 2HUX 18X 2HUY/
SOLV113 1(1112,4E20.7))
SOLV114 2011 FORMAT(35H0 NUMBER OF CYCLES TO CONVERGENCE = 15)
SOLV115 2013 FORMAT(30H1 RESULTS OF ITERATION NO.= 15///)
SOLV116 END

```

```

UNED 1      SUBROUTINE UNED(R,Z,UR,UZ,T,PR,SIGI,CODE,CU,A,P,IX,IFC,JBC)
UNED 2      IMPLICIT REAL*8(A-H,O-Z)
UNED 3      COMMON AA(10000),IA(1500)
UNED 4      COMMON/ONE/ACLER,ACLLZ,TEMP,Q,TOL,VOL,NUMNP,NUMEL,NUMMAT,NUMPC,
UNED 5      *MTYPE,LLL,N,MEAND,NUMBLK,KKK,NCHECK,NP,NL,NLQ,HED(18),NPC
UNED 6      COMMON/TWO/C(3,3),S(10,10),SIG(6),P(8),ST(3,10),RR(5),ZZ(5),XC,YC,
UNED 7      *E(3),LM(4),F(5,4,10),KL(10),NTC(10),MTC(10)
UNED 8      COMMON/THREE/HTGT,NTGT,NIL,M4,STOP
UNED 9      DIMENSION R(NUMNP),Z(NUMNP),UR(NUMNP),UZ(NUMNP),T(NUMNP),PR(NPC,C)
UNED 10     *,SIGI(NUMEL,6),CODE(NUMNP),IX(NUMEL,5),IFC(NPC),JBC(NPC)
UNED 11 C
UNED 12 C
UNED 13      DO 100 I=1,8
UNED 14      P(I)=0.0
UNED 15      DO 100 J=1,8
UNED 16      100 S(I,J)=0.0
UNED 17      MTYPE=IX(N,5)
UNED 18      I=IX(N,1)
UNED 19      J=IX(N,2)
UNED 20      DX=R(J)-R(I)
UNED 21      DY=Z(J)-Z(I)
UNED 22      XL=DSQRT(DX**2+DY**2)
UNED 23      COSA=DX/XL
UNED 24      SINA=DY/XL
UNED 25      COMM=E(1,2,MTYPE)*E(1,4,MTYPE)/XL
UNED 26 C
UNED 27      S(1,1)=COSA*COSA*COMM
UNED 28      S(1,2)=COSA*SINA*COMM
UNED 29      S(1,3)=-S(1,1)
UNED 30      S(1,4)=-S(1,2)
UNED 31      S(2,1)=S(1,2)
UNED 32      S(2,2)=SINA*SINA*COMM
UNED 33      S(2,3)=-S(1,2)
UNED 34      S(2,4)=-S(2,2)
UNED 35      S(3,1)=S(1,3)
UNED 36      S(3,2)=S(2,3)
UNED 37      S(3,3)=S(1,1)
UNED 38      S(3,4)=S(1,2)
UNED 39      S(4,1)=S(1,4)
UNED 40      S(4,2)=S(2,4)
UNED 41      S(4,3)=S(3,4)
UNED 42      S(4,4)=S(2,2)
UNED 43 C
UNED 44 C
UNED 45      RETURN
UNED 46 C
UNED 47      END

```

```

ELEM 1      SUBROUTINE ELEMEN(R,Z,UR,UZ,T,PR,SIG,CODE,CU,A,8,IX,IBC,JBC)
ELEM 2      IMPLICIT REAL*8(A-H,O-Z)
ELEM 3      COMMON AA(10000),IA(1500)
ELEM 4      COMMON/ONE/ACELR,ACELZ,TEMP,Q,TOL,VOL,NUMNP,NUMEL,NUMMAT,NUMPC,
ELEM 5      *MYPE,LLL,N,MEAND,NUMBLK,KKK,NCHECK,NP,NL,NEQ,HED(18),NPC
ELEM 6      COMMON/TWO/C(3,3),S(10,10),SIG(6),P(8),ST(3,10),RR(5),ZZ(5),XC,YC,
ELEM 7      *EE(3),LM(4),E(5,4,10),RO(10),NTC(10),MTC(10)
ELEM 8      COMMON/THREE/MTOT,NTOT,N11,M4,STOP
ELEM 9      DIMENSION R(NUMNP),Z(NUMNP),UR(NUMNP),UZ(NUMNP),T(NUMNP),PR(NPC,2)
ELEM 10     *,SIG(NUMEL,6),CODE(NUMNP),IX(NUMEL,5),IBC(NPC),JBC(NPC)
ELEM 11     DIMENSION CU(NEQ),A(NL,MBAND),8(NEQ)
ELEM 12     DIMENSION U(3),V(3)
ELEM 13 C
ELEM 14     I=IX(N,1)
ELEM 15     J=IX(N,2)
ELEM 16     K=IX(N,3)
ELEM 17     L=IX(N,4)
ELEM 18     MYPE=IX(N,5)
ELEM 19     VOL=0.
ELEM 20     TEMP=(T(I)+T(J)+T(K)+T(L))/4.0
ELEM 21     RATIO=0.0
ELEM 22     NUMTC=NTC(MYPE)
ELEM 23     IF (NUMTC.EQ.1) GO TO 100
ELEM 24     DO 50 M=2,NUMTC
ELEM 25     IF (E(M,1,MYPE)-TEMP) 50,60,60
ELEM 26     50 CONTINUE
ELEM 27     60 DEN=E(M,1,MYPE)-E(M-1,1,MYPE)
ELEM 28     IF (DEN.EQ.0.) GO TO 80
ELEM 29     RATIO=(TEMP-E(M-1,1,MYPE))/DEN
ELEM 30     80 DO 90 KK=1,3
ELEM 31     90 EE(KK)=E(M-1,KK+1,MYPE)+RATIO*(E(M,KK+1,MYPE)-E(M-1,KK+1,MYPE))
ELEM 32     GO TO 110
ELEM 33     100 DO 105 KK=1,3
ELEM 34     105 EE(KK)=E(1,KK+1,MYPE)
ELEM 35     110 COMM=EE(1)/(1.-EE(2)**2)
ELEM 36     C(1,1)=COMM
ELEM 37     C(1,2)=COMM*EE(2)
ELEM 38     C(1,3)=0.
ELEM 39     C(2,1)=C(1,2)
ELEM 40     C(2,2)=C(1,1)
ELEM 41     C(2,3)=0.
ELEM 42     C(3,1)=0.
ELEM 43     C(3,2)=0.
ELEM 44     C(3,3)=.5*COMM*(1.-EE(2))
ELEM 45     DO 130 J=1,10
ELEM 46     DO 120 I=1,3
ELEM 47     120 ST(I,J)=0.
ELEM 48     DO 130 I=1,10
ELEM 49     130 S(I,J)=0.
ELEM 50     DO 140 I=1,4

```

```

ELEM 51      NPP=IX(N,1)
ELEM 52      RR(1)=R(NPP)
ELEM 53      140 ZZ(1)=Z(NPP)
ELEM 54      IF(IX(N,3).EQ.1X(N,4))GO TO 150
ELEM 55      XC=(RR(1)+RR(2)+RR(3)+RR(4))/4.
ELEM 56      YC=(ZZ(1)+ZZ(2)+ZZ(3)+ZZ(4))/4.
ELEM 57      PR(5)=XC
ELEM 58      ZZ(5)=YC
ELEM 59      K=5
ELEM 60      J=1
ELEM 61      I=4
ELEM 62      LM(3)=4
ELEM 63      NT=4
ELEM 64      GO TO 160
ELEM 65      150 NI=1
ELEM 66      LM(5)=5
ELEM 67      I=1
ELEM 68      K=3
ELEM 69      J=2
ELEM 70      XC=(RR(1)+RR(2)+RR(3))/3.
ELEM 71      YC=(ZZ(1)+ZZ(2)+ZZ(3))/3.
ELEM 72      RR(5)=RR(3)
ELEM 73      ZZ(5)=ZZ(3)
ELEM 74      160 DO 200 NN=1,NI
ELEM 75      LM(1)=2*I-1
ELEM 76      LM(2)=2*J-1
ELEM 77      U(1)=ZZ(J)-ZZ(K)
ELEM 78      U(2)=ZZ(K)-ZZ(1)
ELEM 79      U(3)=ZZ(1)-ZZ(J)
ELEM 80      V(1)=RR(K)-PR(J)
ELEM 81      V(2)=RR(1)-RR(K)
ELEM 82      V(3)=RR(J)-RR(1)
ELEM 83      ARFA=(RR(J)*U(2)+RR(1)*U(1)+RR(5)*U(3))/3.
ELEM 84      VOL=VOL+ARFA
ELEM 85      COMM=.25/ARFA
ELEM 86      XNI=NT
ELEM 87      COM=2./XNI
ELEM 88      CUM=CUM*COMM
ELEM 89      DO 180 L=1,3
ELEM 90      II=LM(L)
ELEM 91      ST(1,II)=ST(1,II)+U(L)*CUM
ELEM 92      ST(2,II+1)=ST(2,II+1)+V(L)*COM
ELEM 93      ST(3,II)=ST(3,II)+V(L)*COM
ELEM 94      ST(3,II+1)=ST(3,II+1)+U(L)*CUM
ELEM 95      DO 180 M=1,3
ELEM 96      JJ=LM(M)
ELEM 97      S(11,JJ)=S(11,JJ)+(U(L)*C(1,1)*G(M)+V(L)*C(3,3)*V(M))*COMM
ELEM 98      S(11,JJ+1)=S(11,JJ+1)+(U(L)*C(1,2)*V(M)+V(L)*C(3,3)*U(M))*COMM
ELEM 99      S(11+1,JJ+1)=S(11+1,JJ+1)+(V(L)*C(1,1)*V(M)+U(L)*C(3,3)*U(M))*COMM
ELEM 100     S(JJ+1,11)=S(11,JJ+1)

```



```

ELEM101 180 CONTINUE
ELEM102 I=J
ELEM103 J=J+1
ELEM104 200 CONTINUE
ELEM105 IF(IX(N,3).EQ.IX(N,4))GO TO 250
ELEM106 OU240 I=1,2
ELEM107 KK=10-I
ELEM108 DO 240 K=1,KK
ELEM109 CC=S(KK+1,K)/S(KK+1,KK+1)
ELEM110 DO 230 J=1,3
ELEM111 230 ST(J,K)=ST(J,K)-CC*ST(J,KK+1)
ELEM112 DO 240 J=1,KK
ELEM113 240 S(J,K)=S(J,K)-CC*S(J,KK+1)
ELEM114 250 CONTINUE
ELEM115 IF(LLL.EQ.1) GO TO 400
ELEM116 SIG(1)=-SIGI(N,4)
ELEM117 SIG(2)=-SIGI(N,5)
ELEM118 SIG(3)=-SIGI(N,6)
ELEM119 GO TO 500
ELEM120 C * * * * *
ELEM121 C INITIAL STRESSES ADDED TO THE CALCULATED STRESSES
ELEM122 C * * * * *
ELEM123 400 DT=TEMP-Q
ELEM124 OX=FE(3)*DT
ELEM125 DY=FE(3)*DT
ELEM126 C
ELEM127 C INITIAL STRESSES CALCULATE4 TOTAL HEIGHT =88 FT. AND VALUE OF
ELEM128 C COEFF. KD =0.2 AND UNII WEIGHT = 150 FOR ALL MATERIALS
ELEM129 SIGI(N,2)= -150.0 * (88.0 - YC)
ELEM130 SIGI(N,1)= 0.2* SIGI(N,2)
ELEM131 SIGI(N,3)=0.0
ELEM132 C
ELEM133 SIG(1)=-C(1,1)*DX-C(1,2)*DY+SIGI(N,1)
ELEM134 SIG(2)=-C(2,1)*DX-C(2,2)*DY+SIGI(N,2)
ELEM135 SIG(3)=SIGI(N,3)
ELEM136 C
ELEM137 500 DO 520 I=1,8
ELEM138 P(I)=0.0
ELEM139 DO 510 J=1,3
ELEM140 510 P(I)=P(I)-ST(J,I)*SIG(J)
ELEM141 520 P(I)=P(I)*VOL
ELEM142 IF(LLL.EQ.1) GO TO 540
ELEM143 DO 530 I=1,3
ELEM144 530 SIG(I)=0.0
ELEM145 GO TO 600
ELEM146 540 MM=4
ELEM147 IF(IX(N,3).EQ.IX(N,4)) MM=3
ELEM148 XMM=MM
ELEM149 DY=VOL*ACFLZ*KD(MTYPE)/XMM
ELEM150 DX=VOL*ACELR*KU(MTYPE)/XMM

```

```
ELEM151      DU 550 I=1,MN
ELEM152      P(2*I)=P(2*I)+DY
ELEM153      550 P(2*I-1)=P(2*I-1)+DX
ELEM154      600 RETURN
ELEM155      END
```

```

STRS 1  SUBROUTINE STRESS(R,Z,UR,UZ,T,PR,SIG1,CODE,CU,A,B,IX,IBC,JBC)
STRS 2  IMPLICIT REAL*8(A-H,O-Z)
STRS 3  COMMON AA(10000),IA(1500)
STRS 4  COMMON/ONE/ACELR,ACELZ,TEMP,Q,TOL,VOL,NUMNP,NUMEL,NUMMAT,NUMPC,
STRS 5  *MTYPE,LLL,N,MBAND,NUMBLK,K2K,NCHECK,NP,NL,NFO,HED(16),NPC
STRS 6  COMMON/TWO/C(3,3),S(10,10),SIG(6),P(B),ST(3,10),RR(5),ZZ(5),XC,YC,
STRS 7  *EE(3),LM(4),E(5,4,10),RO(10),NTC(10),MTC(10)
STRS 8  COMMON/THREE/MTOT,NTOT,N11,M4,STOP
STRS 9  DIMENSION R(NUMNP),Z(NUMNP),UR(NUMNP),UZ(NUMNP),T(NUMNP),PR(NPC,2)
STRS 10 * ,SIG1(NUMEL,6),CODE(NUMNP),IX(NUMEL,5),IBC(NPC),JBC(NPC)
STRS 11 DIMENSION CU(NFO),A(NL,MBAND),B(NEQ)
STRS 12 FOR = 0.0
STRS 13 MPRINT=0
STRS 14 DO 600 M=1,NUMEL
STRS 15 N=M
STRS 16 MTYPE=IX(N,5)
STRS 17 SIG1(N,4)=0.
STRS 18 SIG1(N,5)=0.
STRS 19 SIG1(N,6)=0.
STRS 20 IF(IX(N,3).NE.IX(N,2))GO TO 90
STRS 21 I=IX(N,1)
STRS 22 J=IX(N,2)
STRS 23 XC=(R(I)+R(J))/2.0
STRS 24 YC=(Z(I)+Z(J))/2.0
STRS 25 DX=R(J)-R(I)
STRS 26 DY=Z(J)-Z(I)
STRS 27 XL=DSQRT(DX**2+DY**2)
STRS 28 DU=B(2*J-1)-B(2*I-1)
STRS 29 DV=B(2*J)-B(2*I)
STRS 30 DL=DV*DY/XL+DU*DX/XL
STRS 31 SIG(1)=E(1,4,MTYPE)*DL*E(1,2,MTYPE)/XL+SIG1(N,1)*E(1,4,MTYPE)
STRS 32 IF(SIG(1).GT.0.) GO TO 100
STRS 33 SIG1(N,1)=SIG(1)
STRS 34 GO TO 500
STRS 35 100 SIG1(N,4)=E(1,2,MTYPE)*DL/XL+SIG1(N,1)
STRS 36 SIG1(N,1)=0.
STRS 37 GO TO 420
STRS 38 90 CALL ELEMEN(R,Z,UR,UZ,T,PR,SIG1,CODE,CU,A,B,IX,IBC,JBC)
STRS 39 MM=4
STRS 40 IF(IX(N,3).NE.IX(N,4))GO TO 170
STRS 41 MM=3
STRS 42 170 DO 180 I=1,3
STRS 43 RR(I)=0.
STRS 44 DO 180 J=1,MM
STRS 45 II=2*J
STRS 46 JJ=2*IX(N,J)
STRS 47 180 RR(I)=RR(I)+ST(I,II)*B(JJ)+ST(I,II-1)*B(JJ-1)
STRS 48 DO 190 I=1,3
STRS 49 DO 185 J=1,3
STRS 50 185 SIG(1)=SIG(1)+C(I,J)*RR(J)

```

```

STRS 51 190 CONTINUE
STRS 52 IF(LLL.EQ.1) GO TO 195
STRS 53 DO 192 I=1,3
STRS 54 192 SIG(I)=SIG(I)+SIGI(N,I)
STRS 55 195 CONTINUE
STRS 56 CC=(SIG(1)+SIG(2))/2.0
STRS 57 PB=(SIG(1)-SIG(2))/2.0
STRS 58 CR=DSQRT(18**2+SIG(3)**2)
STRS 59 SIG(4)=CC+CR
STRS 60 SIG(5)=CC-CR
STRS 61 SIG(6)=0.0
STRS 62 IF((BE.EQ.0.).AND.(SIG(3).EQ.0.)) GO TO 200
STRS 63 SIG(6)=28.648*ATAN2(SIG(3),BB)
STRS 64 DX=0.0
STRS 65 200 SIGI(N,1)=SIG(1)
STRS 66 SIGI(N,2)=SIG(2)
STRS 67 SIGI(N,3)=SIG(3)
STRS 68 IF((SIG(4).LE.0.).OR.(MTC(MTYPE).EQ.0.)) GO TO 500
STRS 69 IF(SIG(5).GE.0.00001) GO TO 370
STRS 70 EPS=SIG(6)/57.296
STRS 71 CC=DCOS(EPS)
STRS 72 SS=DSIN(EPS)
STRS 73 C2=CC*CC
STRS 74 S2=SS*SS
STRS 75 SC=SS*CC
STRS 76 C
STRS 77 DX=PE(2)*SIG(4)
STRS 78 SIGI(N,4)=SIG(4)*C2+DX*S2
STRS 79 SIGI(N,5)=SIG(4)*S2+DX*C2
STRS 80 SIGI(N,6)=SIG(4)*SC-DX*SC
STRS 81 C
STRS 82 C
STRS 83 GO TO 400
STRS 84 370 SIGI(N,4)=SIG(1)
STRS 85 SIGI(N,5)=SIG(2)
STRS 86 SIGI(N,6)=SIG(3)
STRS 87 400 SIGI(N,1)=SIG(1)-SIGI(N,4)
STRS 88 SIGI(N,2)=SIG(2)-SIGI(N,5)
STRS 89 SIGI(N,3)=SIG(3)-SIGI(N,6)
STRS 90 420 DX=SIGI(N,4)**2+SIGI(N,5)**2+SIGI(N,6)**2
STRS 91 DX=DSQRT(DX)
STRS 92 IF(DX.LE.FUR) GO TO 450
STRS 93 IJK=N
STRS 94 FUR=DX
STRS 95 450 CONTINUE
STRS 96 500 IF(MPRINT.NE.0) GO TO 550
STRS 97 WRITE(6,2000)
STRS 98 MPRINT=50
STRS 99 550 MPRINT=MPRINT+1
STRS100 WRITE(6,2001)N,XC,YC,(SIG(I),I=1,6),DX

```



```

STRS101 600 CONTINUE
STRS102 WRITE(6,2002)FOR,IJK
STRS103 1F(FOR.LE.TOL) NCHECK = 0
STRS104 RETURN
STRS105 2000 FORMAT (7H1EL.NO. 7X 1HX 7X 1HY 4X 8HX-STRESS 4X 8HY-STRESS 3X
STRS106 1 9HXY-STRESS 2X 10HMAX-STRESS 2X 10HMIN-STRESS 7H ANGLE 2X 17HUNP
STRS107 2ALANCED FORCE )
STRS108 2001 FORMAT (17,2F8.2,1P5E12.4,0P1F7.2,1PE20.4)
STRS109 2002 FORMAT(30HOMAXIMUM UNBALANCED FORCE = E12.5,16H IN FLEMENT NO.
STRS110 1 15)
STRS111 END

```

```

MODY 1      SUBROUTINE MODIFY(A,NEQ,MBAND,NI
MODY 2      IMPLICIT REAL*8(A-H,O-Z)
MODY 3      DIMENSION A(NIQ,MBAND)
MODY 4      DO 250 M=2,MBAND
MODY 5      K=N-M+1
MODY 6      IF(K) 235,235,230
MODY 7      230 A(K,M) = 0.0
MODY 8      235 K=N+M-1
MODY 9      IF(NEQ-K) 250,240,240
MODY 10     240 A(N,M) = 0.0
MODY 11     250 CONTINUE
MODY 12     A(N,1)=1.0
MODY 13     RETURN
MODY 14     END

```

```

LOAD 1      SUBROUTINE LOAD(R,Z,UR,UZ,T,PR,SIG1,CODE,CU,A,B,IX,IBC,JBC)
LOAD 2      IMPLICIT REAL*8(A-H,O-Z)
LOAD 3      COMMON AA(10000),IA(1500)
LOAD 4      COMMON/ONE/ACELR,ACELZ,TEMP,Q,TOL,VOL,NUMNP,NUMEL,NUMMAT,NUMPC,
LOAD 5      *MTYPE,LLL,N,MBAND,NUMBLK,KKK,NCHECK,NP,NL,NEQ,HED(16),NPC
LOAD 6      COMMON/TWO/C(3,3),S(10,10),SIG(6),P(8),ST(3,10),RR(5),ZZ(5),XC,YC,
LOAD 7      *EE(3),LM(4),E(5,4,10),RO(10),NTC(10),MTC(10)
LOAD 8      COMMON/THREE/MTOT,NTOT,N11,M4,STOP
LOAD 9      DIMENSION R(NUMNP),Z(NUMNP),UR(NUMNP),UZ(NUMNP),T(NUMNP),PR(NPC,2)
LOAD 10     *,SIG1(NUMEL,6),CODE(NUMNP),IX(NUMEL,5),IBC(NPC),JBC(NPC)
LOAD 11     DIMENSION CU(NEQ),A(NL,MBAND),B(NEQ)
LOAD 12     DO 50 N=1,NUMNP
LOAD 13     B(2*N-1)=UR(N)
LOAD 14     B(2*N)=UZ(N)
LOAD 15     UR(N)=0.
LOAD 16     UZ(N)=0.
LOAD 17     50 CONTINUE
LOAD 18     IF((NUMPC.EQ.0).OR.(LLL.GT.1)) GO TO 300
LOAD 19     DO 200 L=1,NUMPC
LOAD 20     I=IBC(L)
LOAD 21     J=JBC(L)
LOAD 22     DR=Z(I)-Z(J)
LOAD 23     DZ=R(J)-R(I)
LOAD 24     PP2=(PR(L,2)+PR(L,1))/6.
LOAD 25     PP1=PP2+PR(L,1)/6.
LOAD 26     PP2=PP2+PR(L,2)/6.
LOAD 27     II=2*I
LOAD 28     JJ=2*J
LOAD 29     B(II-1)=B(II-1)+PP1*DR
LOAD 30     B(II)=B(II)+PP1*DZ
LOAD 31     B(JJ-1)=B(JJ-1)+PP2*DR
LOAD 32     B(JJ)=B(JJ)+PP2*DZ
LOAD 33     200 CONTINUE
LOAD 34     300 DO 700 N=1,NUMEL
LOAD 35     I=IX(N,1)
LOAD 36     J=IX(N,2)
LOAD 37     K=IX(N,3)
LOAD 38     L=IX(N,4)
LOAD 39     MTYPE=IX(N,5)
LOAD 40     IF(SIG1(N,4).NE.0.) GO TO 320
LOAD 41     IF(SIG1(N,5).NE.0.) GO TO 320
LOAD 42     IF(SIG1(N,6).NE.0.) GO TO 320
LOAD 43     IF(LLL.EQ.1) GO TO 320
LOAD 44     GO TO 700
LOAD 45     320 CONTINUE
LOAD 46     IF(LLL.EQ.1) GO TO 330
LOAD 47     IF(MTC(MTYPE).EQ.0) GO TO 700
LOAD 48     330 IF(J.EQ.K) GO TO 500
LOAD 49     CALL ELEM(R,Z,UR,UZ,T,PR,SIG1,CODE,CU,A,B,IX,IBC,JBC)
LOAD 50     GO TO 600

```

```

LOAD 51 500 CALL UNED(R,Z,UR,UZ,T,PR,SIG1,CODE,CU,A,b,IX,IBC,JBC)
LOAD 52 DX=R(J)-R(1)
LOAD 53 DY=Z(J)-Z(1)
LOAD 54 EP=-SIG1(N,4)/E(1,2,MTYPE)
LOAD 55 DX=DX*EP
LOAD 56 DY=DY*EP
LOAD 57 P(1)=S(1,1)*DX+S(1,2)*DY
LOAD 58 P(2)=S(2,1)*DX+S(2,2)*DY
LOAD 59 P(3)=-P(1)
LOAD 60 P(4)=-P(2)
LOAD 61 600 DO 620 I1=1,4
LOAD 62 620 LM(I1)=2*IX(N,I1)-1
LOAD 63 DO 650 JJ=1,4
LOAD 64 I1=LM(JJ)
LOAD 65 B(I1)=B(I1)+P(2*JJ-1)
LOAD 66 650 B(I1+1)=B(I1+1)+P(2*JJ)
LOAD 67 700 CONTINUE
LOAD 68 DO 750 N=1,NUMNP
LOAD 69 IF(CODE(N).EQ.0.) GO TO 750
LOAD 70 IF((CODE(N).EQ.1.).OR.(CODE(N).EQ.3.)) B(2*N-1)=0.
LOAD 71 IF((CODE(N).EQ.2.).OR.(CODE(N).EQ.3.)) B(2*N)=0.0
LOAD 72 750 CONTINUE
LOAD 73 800 RETURN
LOAD 74 END

```



```

SYMB 1      SUBROUTINE SYMBAN(NG,A,B,NEQ,MBAND,NUMBLK,KKK)
SYMB 2      IMPLICIT REAL*8(A-H,O-Z)
SYMB 3      DIMENSION A(NG,MBAND),B(NEQ)
SYMB 4      NN=NG/2
SYMB 5 C
SYMB 6 C
SYMB 7      NL=NN+1
SYMB 8      NH=NN+NN
SYMB 9      REWIND 1
SYMB 10     GO TO (1000,2000),KKK
SYMB 11     1000 REWIND 2
SYMB 12     NB=0
SYMB 13     GO TO 150
SYMB 14     100 NB=NB+1
SYMB 15     DO 125 N=1,NN
SYMB 16     NM=NN+N
SYMB 17     DO 125 M=1,MBAND
SYMB 18     A(N,M)=A(NM,M)
SYMB 19     125 A(NM,M)=0.
SYMB 20 C
SYMB 21     IF(NUMBLK-NB) 150,200,150
SYMB 22     150 READ (2)((A(N,M),M=1,MBAND),N=NL,NH)
SYMB 23     IF(NB) 200,100,200
SYMB 24     200 DO 300 N=1,NN
SYMB 25     IF(A(N,1).EQ.0.)GO TO 300
SYMB 26     DO 275 L=2,MBAND
SYMB 27     IF(A(N,L).EQ.0.) GO TO 275
SYMB 28     C=A(N,L)/A(N,1)
SYMB 29     I=N+L-1
SYMB 30     J=0
SYMB 31     DO 250 K=L,MBAND
SYMB 32     J=J+1
SYMB 33     250 A(I,J)=A(I,J)-C*A(N,K)
SYMB 34     A(N,L)=C
SYMB 35     275 CONTINUE
SYMB 36     300 CONTINUE
SYMB 37     WRITE(1)((A(N,M),M=1,MBAND),N=1,NN)
SYMB 38     IF(NUMBLK.EQ.NB)GO TO 900
SYMB 39     GO TO 100
SYMB 40     2000 NQ=0
SYMB 41     NB=0
SYMB 42 C
SYMB 43 C
SYMB 44     GO TO 450
SYMB 45     400 NB=NB+1
SYMB 46     DO 425 N=1,NN
SYMB 47     NM=NN+N
SYMB 48     DO 425 M=1,MBAND
SYMB 49     A(N,M)=A(NM,M)
SYMB 50     425 A(NM,M)=0.

```

```

SYMB 51      IF (NUMBLK.EQ.NB)GO TO 500
SYMB 52 450 READ (1) ((A(N,M),M=1,MEAND),N=NL,NH)
SYMB 53      IF (NB.EQ.0)GO TO 400
SYMB 54 500 DO 550 N=1,NN
SYMB 55      J=NQ+N
SYMB 56      DO 540 L=2,MEAND
SYMB 57      I=J+L-1
SYMB 58      IF (NEQ=1) 545,540,540
SYMB 59 540 L(1)=B(1)-A(N,L)*B(J)
SYMB 60 545 IF (A(N,1).EQ.0.) A(N,1)=1.
SYMB 61 550 B(J)=B(J)/A(N,1)
SYMB 62      IF (NUMBLK-NB) 600,650,600
SYMB 63 600 NL=NL+NN
SYMB 64      GO TO 400
SYMB 65 650 BACKSP. CF 1
SYMB 66 700 DO 750 M=1,NN
SYMB 67      N=NN+1-M
SYMB 68      J=NQ+N
SYMB 69      DO 750 L=2,MEAND
SYMB 70      IF (A(N,L).EQ.0.)GO TO 750
SYMB 71      I=J+L-1
SYMB 72      IF (NEQ.LT.1)GO TO 750
SYMB 73      B(J)=B(J)-A(N,L)*B(1)
SYMB 74 750 CONTINUE
SYMB 75      NB=NB-1
SYMB 76      IF (NB.EQ.0) GO TO 900
SYMB 77      BACKSPACE 1
SYMB 78      DO 800 N=1,NN
SYMB 79      NM=NN+N
SYMB 80      DO 800 M=1,MEAND
SYMB 81      A(NM,M)=A(N,M)
SYMB 82 800 A(N,M)=0.
SYMB 83      READ(1) ((A(N,M),M=1,MEAND),N=1,NN)
SYMB 84      BACKSPACE 1
SYMB 85      NL=NQ-NN
SYMB 86      GO TO 700
SYMB 87 900 RETURN
SYMB 88      END

```

CHAPTER II: ELPL - A Computer Program for Plane Strain Analysis of Stresses, Deformations and Progressive Failure in Elastic-Perfectly Plastic Rock

2.1. Purpose and Capability

This program is applicable to plane strain analysis of stresses, deformations and progressive failure in elastic-perfectly plastic material following generalized Mohr-Coulomb yield criterion, and the incremental theory of plasticity. Arbitrary initial stresses, arbitrary sequence of excavation and construction, arbitrary history of load application can be simulated. One-dimensional elements with prescribed prestress can be included in the analysis. The computer program is applicable to study of stability of underground or surface excavations, evaluation of alternative schemes for excavation, comparative study of support stresses and deformation.

Theoretical development incorporated in the program is discussed in Part I- Technical Report, of this report.

2.2. Program Organization

The computer program is in Fortran language. Tapes 1 and 2 are used as scratch files. Tapes 5 and 6 are the input/output files. The listing in section 2.5 uses double precision for real numbers. The program capacity can be altered by changing the dimension of arrays AA and IA. These correspond to the total locations required in double precision real arrays and integer arrays respectively. NTOT, MTOT at lines MAIN 28, MAIN 27 are set equal to the dimension of AA and IA.

The program consists of the following units.

a. MAIN

In this unit, the control information including maximum number of elements, nodal points, different materials, boundary pressure cards, the number of steps of excavation or construction, the maximum number of elements removed from the system or added to it at any step, is read in. This information is used to organize the dimensions of various arrays in the analysis. This done, the analysis moves on to the next unit.

b. Subroutine INPT

This subroutine is called by the unit MAIN. The first step is to read in material property data, for all the different materials that may participate in the system at any stage. Also nodal point coordinates, loads and code on boundary conditions are read in or generated. Element geometry, initial stresses and element thickness is also read in or generated. The initial stresses may be input or computed within the program. The thickness, if not specified, is assumed to be unity. Maximum bandwidth for the system is calculated and dimensions of blocks for generation and storage of the system stiffness matrix are defined. After defining these controls, the incremental structure is analyzed in steps. For each step the number of nodal points, the number of elements, the number of elements removed or added, if any, the number of boundary pressure cards, the material type of the elements added or changed in material properties, the number of nodal points removed or added to the system in the step considered, the thickness of elements added, are read in. Additional information defines whether a step involves excavation or construction. At any step, several changes can be simultaneously introduced into a structure.

For example, addition of several sets of elements of different materials, changes in material properties of other elements, removal of certain elements can all be introduced simultaneously if desired. For each step the boundary pressures are read in. After all the information is assembled, the solution process is transferred to subroutine SOLVE.

e. Subroutine SOLVE

This subroutine called by INPT is concerned with obtaining the stresses and deformations at a given stage of the incremental structure allowing for progressive failure. To trace the progressive failure, the solution process traces a sequence of elements reaching yield under the loads applied. This sequence of yield is associated with a proportion of the load application and is described in the output as successive approximations with increasing 'stress ratio.' The procedure consists of applying the total load and then scaling it according to the minimum rates of load increment needed to ensure an excursion to yield by one element at a time.

The SOLVE subroutine calls ONED and QUAD to obtain stiffness of one-dimensional and two-dimensional elements respectively. This is added to system stiffness which is stored on tape after modification for prescribed displacement boundary conditions. Solution to the stiffness equations is obtained in subroutine BANSOL and subroutine STRESS defines the stresses, the stress ratio, the scaling of stress increments and the control for continuation of progressive failure analysis.

d. Subroutine ONED

This subroutine generates stiffness of one-dimensional elements as well as the

forces corresponding to the unbalanced stress defined by the difference in total load application and the load taken by the system in the current approximation in progressive failure analysis.

c. Subroutine QUAD

This subroutine generates the stiffness for two-dimensional elements (quadrilaterals or triangles). For the current load increment, an element is either elastic or has reached yield and is plastic. The subroutine STRSTR is called to obtain the stress-strain relationship. Gravity loads, and loads corresponding to unbalanced stress are generated in this subroutine.

f. Subroutine STRSTR

This subroutine is called by QUAD and also by STRESS. It defines the stress-strain relationship for the elastic or plastic materials as the case may be. For the elastic case the relationship is

$$\begin{pmatrix} \sigma_x \\ \sigma_y \\ \tau_{xy} \end{pmatrix} = \frac{E^*}{1 - \nu^{*2}} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1 - \nu^*}{2} \end{bmatrix} \begin{pmatrix} \epsilon_x \\ \epsilon_y \\ \gamma_{xy} \end{pmatrix}$$

and $\sigma_z = \nu (\sigma_x + \sigma_y)$

where $E^* = \frac{E}{1 - \nu^2}$, $\nu^* = \frac{\nu}{1 - \nu}$

and E, ν are respectively the Young's modulus and Poisson's ratio for the isotropic elastic material. If anisotropy is to be considered, the above relationship can be modified to reflect that property.

For the plastic domain the stress-strain relation is

$$\begin{Bmatrix} \dot{\sigma}_x \\ \dot{\sigma}_y \\ \dot{\tau}_{xy} \\ \dot{\sigma}_z \end{Bmatrix} = \begin{bmatrix} D_{11} & D_{12} & D_{13} \\ D_{21} & D_{22} & D_{23} \\ D_{31} & D_{32} & D_{33} \\ D_{41} & D_{42} & D_{43} \end{bmatrix} \begin{Bmatrix} \dot{\epsilon}_x \\ \dot{\epsilon}_y \\ \dot{\gamma}_{xy} \end{Bmatrix}$$

$$\text{where } D_{11} = 2G (1 - h_2 - 2h_1 \sigma_x - h_3 \sigma_x^2)$$

$$D_{21} = D_{12} = -2G (h_2 + h_1 (\sigma_x + \sigma_y) + h_3 \sigma_x \sigma_y)$$

$$D_{31} = D_{13} = -2G (h_1 \tau_{xy} + h_3 \sigma_x \tau_{xy})$$

$$D_{22} = 2G (1 - h_2 - 2h_1 \sigma_y - h_3 \sigma_y^2)$$

$$D_{32} = D_{23} = -2G (h_1 \tau_{xy} + h_3 \sigma_y \tau_{xy})$$

$$D_{33} = 2G (.5 - h_3 \tau_{xy}^2)$$

$$D_{41} = -2G (h_2 + h_1 (\sigma_x + \sigma_z) + h_3 \sigma_x \sigma_z)$$

$$D_{42} = -2G (h_2 + h_1 (\sigma_y + \sigma_z) + h_3 \sigma_y \sigma_z)$$

$$D_{43} = -2G (h_1 \tau_{xy} + h_3 \tau_{xy} \sigma_z)$$

$$2G = \frac{E}{(1 + \nu)}$$

$$h_1 = \frac{.5 h_4}{h_5 J_2^{\frac{1}{2}}}$$

$$h_2 = \frac{h_4 h_6}{h_5} - \frac{\nu}{(1 - 2\nu)} - \frac{k}{h_5 J_2^{\frac{1}{2}}}$$

$$h_3 = \frac{.5}{h_5 J_2}$$

$$h_4 = 3\alpha \frac{K}{G} - \frac{J_1'}{3 J_2'^2}$$

$$h_5 = 1 + 9\alpha^2 \frac{K}{G}$$

$$h_6 = \frac{J_1'}{J_2'^2}$$

$$K = \text{bulk modulus} = \frac{E}{3(1 - 2\nu)}$$

$$J_1' = \sigma_x + \sigma_y + \sigma_z = \text{first invariant of the stress tensor}$$

$$J_2' = \text{the second invariant of the stress deviation tensor.}$$

g. Subroutine MODIFY

This subroutine is called by the subroutine SOLVE to modify the stiffness matrix for prescribed displacement boundary conditions. The modified matrix is returned to SOLVE.

h. Subroutine BANSOL

This subroutine called by the subroutine SOLVE solves the stiffness equations by gaussian elimination using auxiliary storage files 1 and 2. Results are stored in B array and returned to SOLVE.

i. Subroutine STRESS

This subroutine is called by the subroutine SOLVE after displacements corresponding to a load increment have been computed. As a first step the entire load is assumed to be applied and the resulting stress state checked for possible excursion beyond yield. If the total load application shows an element passing from the elastic to the plastic stage, the stress ratio is computed as explained in Part I-Technical Report. The minimum stress ratio of all elements corresponds to the least load incre-

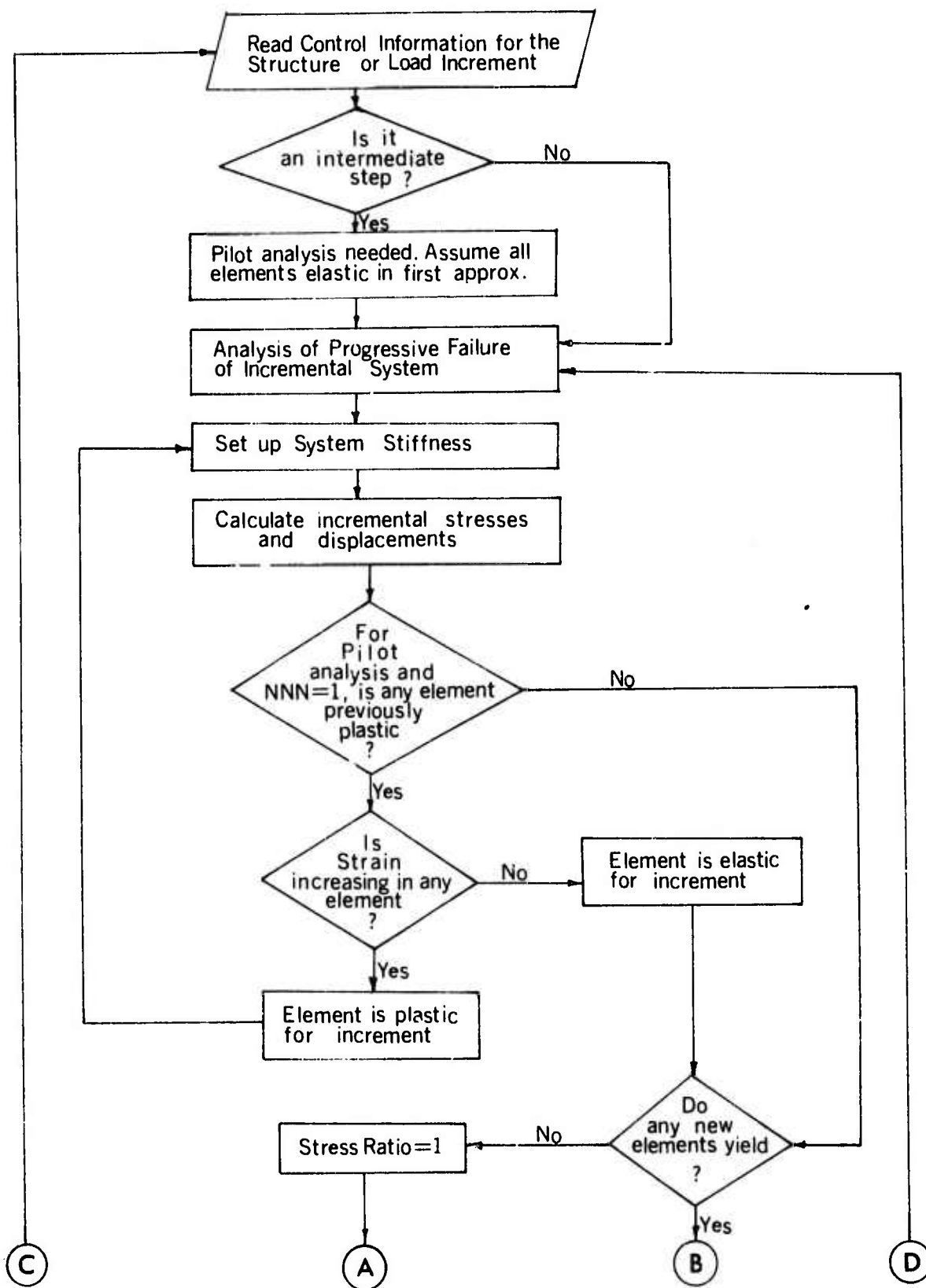
ment necessary for at least one more element to pass from elastic to the plastic range. The stress ratio also has to be such that elements previously on the yield surface do not depart from this surface by more than the designed tolerance. To avoid unnecessarily lengthy computation, the ratio is kept above a certain minimum. Elements yielding within a small fraction of the load corresponding to the stress ratio are assumed also to have become plastic for the purpose of subsequent computations.

The stiffness for elements in plastic range is stress dependent. Thus over an increment of load the stiffness will change. Noting that plastic yield is, in general, local in character, it is reasonable to assume that the change in stiffness will affect the stress without significantly altering the strains or displacements. An iterative procedure is included to allow for this. Also to determine whether elements initially in plastic range unload elastically in a non-monotonic loading sequence, a pilot analysis is carried out, assuming all elements to be elastic, whenever the structure geometry is changed or a new loading applied. Details of these procedures are discussed in Part I-Technical Report.

After evaluation of stress ratio, the correct stresses in all elements are calculated corresponding to yield of the next group of elements. The difference between the incremental stress corresponding to the total load increment and the stress allowed by the stress ratio is treated as unbalanced stress and applied as pseudo-load in the next iteration.

2.3. Input Data

Input to the program is a sequence of punched cards in the following order and format:



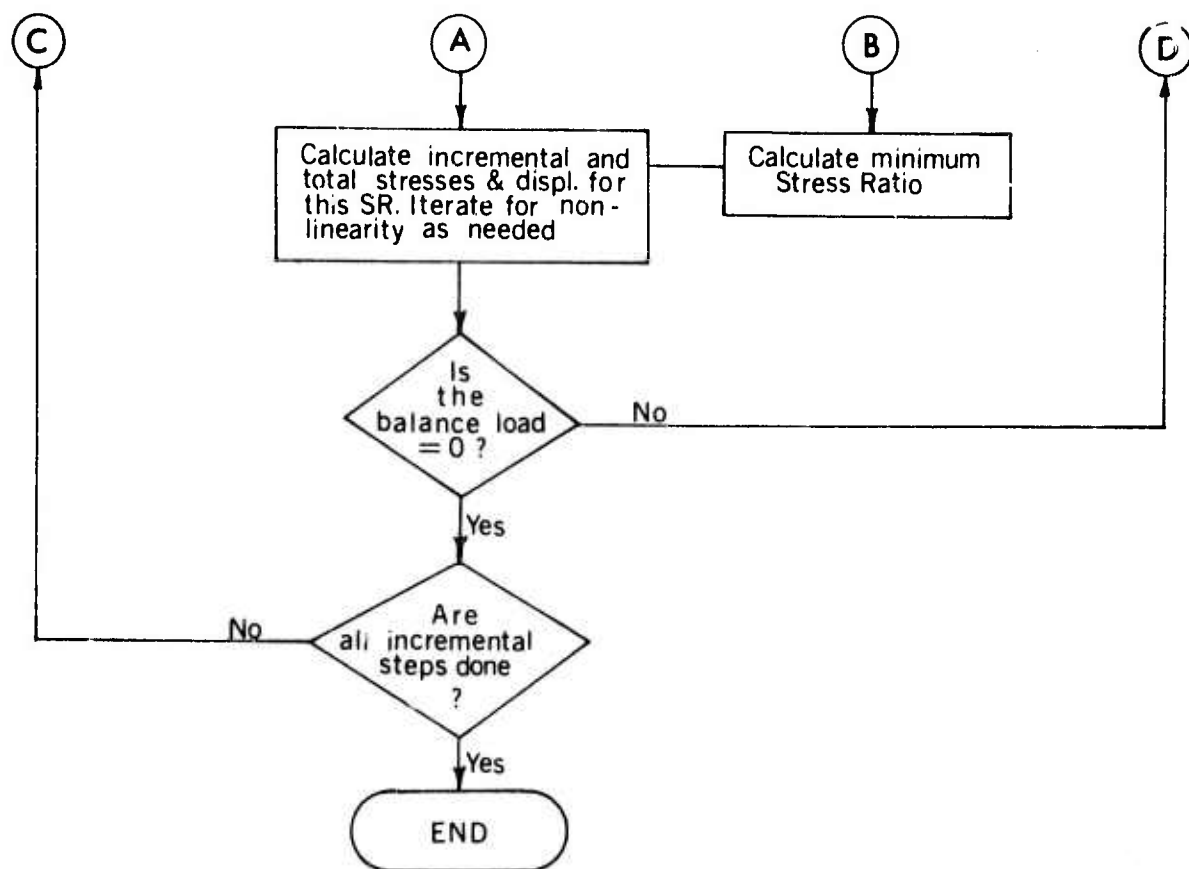


FIG. II-1. Flow Chart for Progressive Failure of Elastic-Plastic Incremental System

a. First Card. Job Title (18A4)

This card carries a descriptive title to identify the job.

b. Second Card. Control Information (4I5, 2F10.2, 3I5)

This card carries the control information for the problem being solved.

<u>Information</u>	<u>Columns</u>
Total number of nodal points (NUMNP)	1-5
Total number of elements (NUMEL)	6-10
Number of different materials (NUMMAT)	11-15
Number of pressure cards (NUMPC)	16-20
Acceleration in X-direction (ACELR)	21-30
Acceleration in Y-direction (ACELZ)	31-45
Total number of excavation and construction steps (NSTEP)	46-50
Code to designate whether initial stress is being fed as data or will be evaluated before the first step (MCASE)	51-55
MCASE = 0, initial stress will be evaluated = 1, initial stress is fed as data	
Maximum number of elements/nodal points removed or added in any one step (NMR).	56-60

c. Material property cards

One set of cards will be provided for each material. Each set will consist of the following cards:

i. First Card (I5, F10.0)

<u>Information</u>	<u>Columns</u>
Material identification number (MTYPE)	1-5
Mass density of material (RO)	6-15

ii. Second Card (5F10.0) for two-dimensional element

<u>Information</u>	<u>Columns</u>
Elastic modulus	1-10
Poisson's ratio	11-20
Cohesion	21-30
Angle of internal friction in degrees	31-40

or

iii. Second card (5F10.0) for one-dimensional element

<u>Information</u>	<u>Columns</u>
Elastic modulus	1-10
Poisson's ratio	11-20
Code = 1, if element is prestressed = 0, if element is not prestressed	21-30
Allowable compressive strength of the material if the element has prestressing (code = 1)	31-40
Area of one-dimensional element	41-50

d. Nodal Point Data (I5, F5.0, 5 F10.0)

One card for each nodal point with the following information is provided.

<u>Information</u>	<u>Columns</u>
Nodal point number	1-5
Type of nodal point (CODE)	6-10
X-ordinate	11-20

Y-ordinate	21-30
X load or displacement (XR)	31-40
Y load or displacement (XZ)	41-50

The code will be defined as follows:

<u>CODE</u>	<u>Implication</u>
0.0	XR is the specified X-load and XZ is the specified Y-load
1.0	XR is the specified X-displacement and XZ is the specified Y-load
2.0	XR is the specified X-load and XZ is the specified Y-displacement
3.0	XR is the specified X-displacement and XZ is the specified Y-displacement.

Nodal point cards must be in numerical sequence. Nodal points for which no cards are input will be generated by interpolation between specified nodal points. These points will have CODE and loads set equal to zero. The X, Y coordinates will be linearly interpolated.

e. Element Data (6I5, 5F10.0)

One card for each element, in numerical sequence, will show:

<u>Information</u>	<u>Columns</u>
Element number	1-5
Nodal point I	6-10
Nodal point J	11-15
Nodal point K	16-20
Nodal point L	21-25
Material type	26-30

Nodal points are labelled I, J, K, L counter-clockwise.

Initial stresses:

Component in X-direction	31-40
Component in Y-direction	41-50
Shearing stress in X-Y plane	51-60
Component in Z-direction	61-70
Thiekness of element (TH)	71-80

If the eolumns for thiekness are left blank, thiekness will be taken as 1.0.

Elements omitted from the sequenee will be generated. The material type and thickness for generated elements is the same as for the preeeding element.

f. If the initial stresses are to be evaluated (i.e. if MCASE = 0 in eard b), the following card should be included; otherwise proceed to g.

Initial Stress Evaluation

i. First eard (18A4)

This eard gives the descriptive title of the step.

ii. Second card (4I5)

<u>Information</u>	<u>Columns</u>
Number of nodal points in this step	1-5
Number of elements in this step	6-10
Blank	11-15
Number of pressure boundary eards	16-20

g. Incremental Step Information

One set of cards will be provided for each step of construction or dismantling as follows:

i. First card (18A4)

This card will give a descriptive title of the step.

ii. Second card (10I5)

<u>Information</u>	<u>Columns</u>
Number of nodal points in this step	1-5
Number of elements in this step	6-10
Number of elements removed/added	11-15
Number of pressure boundary cards	16-20
Material type of new elements	21-25
Code = 0 for dismantling = 1 for construction	26-30
Number of nodal points removed/added	31-35

If in addition to adding/removing elements, it is desired to change the material type of some existing elements, use columns 36-45.

Number of elements for which material type is to be changed	36-40
New material type of changed elements	41-45

If it is desired to add/remove two material types in one step, use columns 46-50.

KMORE = 1, another material is being added/removed = 0, no other material is being added/removed.	46-50
--	-------

iii. Elements removed/added (16I5)

One or more cards will indicate the element numbers removed or added in this step. Total number of elements should be the same as in columns 11-15 in card ii.

iv. Nodal points removed/added (16I5)

One or more cards will indicate the nodal point numbers removed or added in this step. The total number should be the same as in columns 31-35 of card ii.

v. Elements for which material type changed (16I5)

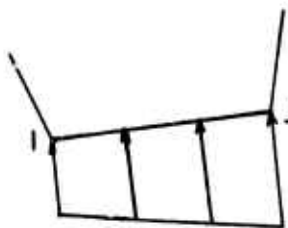
One or more cards will indicate the element numbers for which material type has been changed. Total number should be the same as in columns 36-40 in card ii.

If KMORE = 1, repeat card i to v for the next material; otherwise proceed to card vi.

vi. Pressure boundary cards (2I5, 2F10.0)

The cards representing pressure at the boundary will be as follows:

<u>Information</u>	<u>Columns</u>
Nodal point I	1-5
Nodal point J	6-10
Total normal pressure at I	11-20
Total normal pressure at J	21-30



As shown in the sketch, the boundary of the element must be on the left hand side as one progresses from I to J. Surface tensile force is input as a negative pressure.

2.4. Output information

The following information is developed and printed by the program:

- a. Reprint of the input data
- b. Cumulative nodal point displacement after each step
- c. Stressses at the center of each element after each step
- d. Stress ratios for elements which can yield with full application of the balance load
- e. The strss ratio applicable for any iteration and the consideration leading to its determination c.g. yielding of next element, minimum stress ratio, maximum stress ratio, ensuring that elements previously in yield surface do not move away from that surface beyond a specified tolerance.
- f. The proportion of total load increment applied upto the end of current iteration.
- g. The next group of elements reaching yield
- h. The failure ratio for each element.

2.5. Fortran Listing

```

MAIN 1 C * FINITE ELEMENT ANALYSIS OF ELASTIC-PERFECTLY PLASTIC *
MAIN 2 C * MOHR-COULOMB SOLIDS UNDER PLANE STRAIN. INCREMENTAL *
MAIN 3 C * CONSTRUCTION EXCAVATION AND INITIAL STRESSES ARE *
MAIN 4 C * CONSIDERED. A 4-CST QUADRILATERAL ELEMENT IS USED. *
MAIN 5 C * THE FORMULATION IS DOCUMENTED IN THE FINAL REPORT *
MAIN 6 C * ON CONTRACT HO210017 BETWEEN THE OHIO STATE UNIVERSITY *
MAIN 7 C * AND THE UNITED STATES BUREAU OF MINES SUPPORTED BY *
MAIN 8 C * THE ADVANCED RESEARCH PROJECTS AGENCY. *
MAIN 9 C * PROGRAMMERS: R.D.SINGH AND K.J.SINGH *
MAIN 10 IMPLICIT REAL*8(A-H,O-Z)
MAIN 11 COMMON/TOTAL/ AA(30000), IA(7000)
MAIN 12 COMMON/ONE/NUMNP,NUMEL,NUMMAT,NUMPC,NPMAX,NELMAX,N,NNN,KKK,KLK,
MAIN 13 * ISHIFT,MBAND,NBAND,NUMBLK,MTYPE,NCASE,MCASE,NMR,NL,NPP,NFQ,NPC
MAIN 14 * ,NP,NSTEP,NCODE
MAIN 15 COMMON/TWO/PP(5),ZZ(5),S(10,10),P(10),ST(3,10),C(4,4),SIG(7),FF(4)
MAIN 16 * ,TITLE(18),SR1,SR2,XC,YC,ACELR,ACELZ,BCELR,BCELZ,VOL
MAIN 17 COMMON/THREE/STOP,MTOT,NTOT,N13,N14,MB
MAIN 18 C
MAIN 19 C ACELR,ACELZ ARE BODY FORCE INTENSITIES. MCASE IS INPUT AS
MAIN 20 C ZERO IF INITIAL STRESSES HAVE TO BE COMPUTED AS PART OF
MAIN 21 C THE ANALYSIS.
MAIN 22 C
MAIN 23 READ(5,1000)TITLE ,NUMNP,NUMEL,NUMMAT,NUMPC,ACELR,ACELZ,
MAIN 24 * ,NP,NSTEP,MCASE,NMR
MAIN 25 WRITE (6,2000) TITLE,NUMNP,NUMEL,NUMMAT,NUMPC,ACELR,ACELZ,NP,
MAIN 26 * ,NSTEP,NMR
MAIN 27 MTOT=7000
MAIN 28 NTOT=30000
MAIN 29 NPC=NUMPC
MAIN 30 IF(NUMPC.EQ.0) NPC=1
MAIN 31 IF(NMR.EQ.0) NMR=1
MAIN 32 N1=1
MAIN 33 N2=N1+NUMNP
MAIN 34 N3=N2+NUMNP
MAIN 35 N4=N3+NUMNP
MAIN 36 N5=N4+NUMNP
MAIN 37 N6=N5+NUMNP
MAIN 38 N7=N6+2*NUMNP
MAIN 39 N8=N7+5*NUMMAT
MAIN 40 N9=N8+NUMMAT
MAIN 41 N10=N9+2*NPC
MAIN 42 N11=N10+8*NUMEL
MAIN 43 N12=N11+NUMFL
MAIN 44 N13=N12+NUMEL
MAIN 45 M1=1
MAIN 46 M2=M1+5*NUMFL
MAIN 47 M3=M2+NUMFL
MAIN 48 M4=M3+NPC
MAIN 49 M5=M4+NPC
MAIN 50 M6=M5+NMR

```

```

MAIN 51      M7=M6+NMR
MAIN 52      MR=M7+NMR
MAIN 53      JJ=MR-MTOT
MAIN 54      IF (JJ.LE.0) GO TO 100
MAIN 55      WRITE(6,3000)JJ
MAIN 56      CALL EXIT
MAIN 57      100 CONTINUE
MAIN 58      NEO=2*NUMNP
MAIN 59      CALL INPT (AA(N1),AA(N2),AA(N3),AA(N4),AA(N5),AA(N6),AA(N7),
MAIN 60      *          AA(N8),AA(N9),AA(N10),AA(N11),AA(N12), 1A(M1),1A(M2),
MAIN 61      *          1A(M3),1A(M4),1A(M5),1A(M6),1A(M7) 1
MAIN 62      1000 FORMAT(18A4/4I5,2F10.2,4I5)
MAIN 63      2000 FORMAT(1H1,18A4/
MAIN 64      .30H0 NUMBER OF NODAL POINTS 110/
MAIN 65      .30H0 NUMBER OF ELEMENTS 110/
MAIN 66      .30H0 NUMBER OF DIFF. MATERIALS 110/
MAIN 67      .30H0 NUMBER OF PRESSURE CARDS 110/
MAIN 68      .29H0 X-ACCELERATION F11.1/
MAIN 69      .29H0 Y-ACCELERATION F11.1/
MAIN 70      .30H0 NUMBER OF APPROXIMATIONS 110/
MAIN 71      .30H0 NUMBER OF STEPS 110/
MAIN 72      .69H0 MAX. NO. OF NODAL POINTS OR ELEMENTS ADDED OR REMOVED AT ANY
MAIN 73      . STEP 15)
MAIN 74      3000 FORMAT(70H EXECUTION TERMINATED IN MAIN PROGRAM. REQUIRED CORR BY
MAIN 75      .CEEDS MTOT BY 110)
MAIN 76      STOP
MAIN 77      END

```



```

INPT 1      SUBROUTINE INPT (R,Z,UR,UZ,CODE,BU,F,RO,PR,SIGI,TH,FR,IX,MTAG,
INPT 2      *      IRC,JBC,NUMR,NUMRI,NNP)
INPT 3      IMPLICIT REAL*8(A-H,O-Z)
INPT 4      COMMON/TOTAL/ AA(30000), IA(7000)
INPT 5      COMMON/ONE/NUMNP,NUMEL,NUMMAT,NUMPC,NPMAX,NFLMAX,N,NNN,KKK,CLK,
INPT 6      *      ISHIFT,MBAND,NBAND,NUMPLK,MTYPE,NCASF,MCASF,NMR,NL,NPP,NEQ,NPC
INPT 7      *      ,NP,NSTEP,NCODE
INPT 8      COMMON/TWO/RR(5),ZZ(5),S(10,10),P(10),ST(3,10),C(4,4),SIG(7),EF(4)
INPT 9      *      ,TITLE(10),SR1,SR2,XC,YC,ACFLR,ACELZ,RCFLR,RCFLZ,VOL
INPT 10     COMMON/THREE/STOP,MTOT,NTOT,N13,N14,MR
INPT 11     DIMENSION R(NUMNP),Z(NUMNP),UR(NUMNP),UZ(NUMNP),CODE(NUMNP),
INPT 12     *      BO(NEQ),E(5,NUMMAT),RO(NUMMAT),PR(NPC,2),SIGI(NUMEL,8),
INPT 13     *      TH(NUMEL),FR(NUMEL),IX(NUMEL,5),MTAG(NUMEL),IRC(NPC),
INPT 14     *      JBC(NPC),NUMR(NMR),NUMRI(NMP),NNP(NMR)
INPT 15 C
INPT 16 C      INPUT MATERIAL PROPERTIES
INPT 17 C
INPT 18      DO 55 M=1,NUMMAT
INPT 19      READ (5,1001) MTYPE,RO(MTYPE)
INPT 20      WRITE(6,2001) MTYPE,RO(MTYPE)
INPT 21      READ (5,1002) (E(J,MTYPE),J=1,5)
INPT 22      WRITE(6,2002) (E(J,MTYPE),J=1,5)
INPT 23      55 CONTINUE
INPT 24 C
INPT 25 C      READ AND PRINT NODAL POINT DATA
INPT 26 C
INPT 27      WRITE (6,2003)
INPT 28      L=0
INPT 29      60 READ (5,1003) N,CODE(N),R(N),Z(N),UR(N),UZ(N)
INPT 30      NL=L+1
INPT 31      IF (N.EQ.1) GO TO 70
INPT 32      ZX=N-L
INPT 33      DR=(R(N)-R(L))/ZX
INPT 34      DZ=(Z(N)-Z(L))/ZX
INPT 35      70 L=L+1
INPT 36      IF(N-L) 100,90,80
INPT 37      80 CODE(L)=0.0
INPT 38      R(L)=R(L-1)+DR
INPT 39      Z(L)=Z(L-1)+DZ
INPT 40      UR(L)=0.0
INPT 41      UZ(L)=0.0
INPT 42      GO TO 70
INPT 43      90 IF(NUMNP-N) 100,110,60
INPT 44      100 WRITE (6,2005) N
INPT 45      CALL EXIT
INPT 46      110 WRITE (6,2004)((K,CODE(K),R(K),Z(K),UR(K),UZ(K)),K=1,NUMNP)
INPT 47 C
INPT 48 C      READ AND PRINT ELEMENT DATA
INPT 49 C
INPT 50      WRITE (6,2006)

```

```

INPT 51      N=0
INPT 52      130 READ (5,1004) M,(IX(M,1),I=1,5),(SIGI(M,1),I=1,4),TH(M)
INPT 53      IF (TH(M).EQ.0.) TH(M)=1.0
INPT 54      ZX=M-N
INPT 55      IF (N.EQ.0) GO TO 140
INPT 56      DO 135 I=1,4
INPT 57      135 SIG(I)=(SIGI(M,1)-SIGI(N,1))/ZX
INPT 58      140 N=N+1
INPT 59      IF (M.LE.N) GO TO 170
INPT 60      IX(N,1)=IX(N-1,1)+1
INPT 61      IX(N,2)=IX(N-1,2)+1
INPT 62      IX(N,3)=IX(N-1,3)+1
INPT 63      IX(N,4)=IX(N-1,4)+1
INPT 64      IX(N,5)=IX(N-1,5)
INPT 65      DO 160 I=1,4
INPT 66      160 SIGI(N,1)=SIGI(N-1,1)+SIG(I)
INPT 67      TH(N)=TH(N-1)
INPT 68      170 IF (M.GT.N) GO TO 140
INPT 69      IF (NUMEL.GT.N) GO TO 130
INPT 70      WRITE(6,2007) ((N,(IX(N,1),I=1,5),(SIGI(N,1),I=1,4),TH(N)),N=1,NUM
INPT 71      .      FL)
INPT 72 C
INPT 73 C      INITIALIZATION OF UNBALANCED STRESSES AND CUMULATIVE
INPT 74 C      DISPLACEMENTS
INPT 75 C
INPT 76      DO 195 N=1,NUMEL
INPT 77      MTA6(N)=0
INPT 78      SIGI(N,5)=0.
INPT 79      SIGI(N,6)=0.
INPT 80      SIGI(N,7)=0.
INPT 81      195 SIGI(N,8)=0.
INPT 82      DO 200 N=1,NUMNP
INPT 83      RO(2*N-1)=0.
INPT 84      200 RO(2*N)=0.
INPT 85 C
INPT 86 C      CALCULATE MAXIMUM BAND-WIDTH FOR THE INCREMENTAL SYSTEM
INPT 87 C
INPT 88      J=0
INPT 89      DO 250 N=1,NUMEL
INPT 90      DO 250 I=1,4
INPT 91      DO 250 L=1,4
INPT 92      KK=IABS(IX(N,1)-IX(N,L))
INPT 93      IF (KK.GT.J) J=KK
INPT 94      250 CONTINUE
INPT 95      NBAND=2*J+2
INPT 96      WRITE(6,3000) NBAND
INPT 97 C
INPT 98 C      CALCULATE BLOCKSIZE CONSISTENT WITH AVAILABLE CORE
INPT 99 C
INPT 100      NL=(NTOT-N13+1)/(NBAND+1)

```

```

INPT101      NLL=NFQ+3
INPT102      IF(NL.GT.NLL) NL=NLL
INPT103      NL=NL/4
INPT104      NL=4*NL
INPT105      N14=N13+NL
INPT106      NB=2*NBAND
INPT107      NZ=N13+NB+NE*NBAND-I
INPT108      WRITE (6,4010) NZ
INPT109      IF(NZ.GT.NTOT) CALL EXIT
INPT110      IF (NL.L1.NE) CALL FXIT
INPT111 C     .....
INPT112 C     . READ CONTROL INFORMATION FOR THE NEXT STEP IN INCREMENTAL .
INPT113 C     . ANALYSIS .
INPT114 C     .....
INPT115      NCASF=0
INPT116      IF(MCASE.NE.0) NCASF=1
INPT117      IF(MCASE.NE.0) MCASF=1
INPT118 300 READ(5,1008) TITLE
INPT119      WRITE(6,2011) TITLE
INPT120      READ(5,1009) NPMAX,NELMAX,NUMFR,NUMPC,MTYPE,NCODE,NPMIS,
INPT121      .      NUMFR1,MTYPE1,KMORF,THICK
INPT122      IF(THICK.EQ.0.) THICK=1.0
INPT123      IF(NUMFR.EQ.0) GO TO 310
INPT124      READ(5,1007) (NUMR(N), N= 1,NUMFR)
INPT125      DO 305 I=1,NUMFR
INPT126      NUM=NUMR(I)
INPT127      TH(NUM)=THICK
INPT128 305 CONTINUE
INPT129 310 IF(NPMIS.EQ.0) GO TO 320
INPT130      READ(5,1007) (NNP(M), M = 1,NPMIS)
INPT131 320 IF (NUMFR1.EQ.0) GO TO 330
INPT132      READ(5,1007) ( NUMR1(N),N=1,NUMFR1)
INPT133 C
INPT134 C     DETERMINE BAND-WIDTH FOR THIS STEP
INPT135 C
INPT136 330 J=0
INPT137      IF(KMORF.NE.0) GO TO 342
INPT138      DO 340 N = 1,NELMAX
INPT139      DO 340 I=1,4
INPT140      DO 340 L=1,4
INPT141      KK=IARS(IX(N,))-IX(N,L))
INPT142      IF(KK.GT.J) J=KK
INPT143 340 CONTINUE
INPT144      MBAND=2*J+2
INPT145 342 IF(NPMIS.EQ.0) GO TO 346
INPT146      DO 345 I = 1, NPMIS
INPT147      J=NNP(I)
INPT148      RD(2*J-1)=0.
INPT149      BD(2*J)=0.
INPT150 345 CONTINUE

```

```

INPT151 C
INPT152 C      PRINT CONTROL INFORMATION FOR THE CURRENT STEP IN INCREMENTAL
INPT153 C      ANALYSIS
INPT154 C
INPT155 346 IF(NCODE.FQ.1) GO TO 410
INPT156      IF(NUMER.FQ.0) GO TO 351
INPT157      WRITE(6,2012)INCASF,NPMAX,NELMAX,NUMPC,NPMIS,MTYPE,
INPT158      (NUMR(N),N=1,NUMER)
INPT159      GO TO 352
INPT160 351 WRITE(6,2012)INCASF,NPMAX,NELMAX,NUMPC,NPMIS,MTYPE
INPT161 352 IF(NPMIS.FQ.0) GO TO 420
INPT162      WRITE(6,2014) (NNP(M),M=1,NPMIS)
INPT163      GO TO 420
INPT164 410 IF(NUMER.FQ.0) GO TO 411
INPT165      WRITE(6,2013)INCASF,NPMAX,NELMAX,NUMPC,NPMIS,MTYPE,
INPT166      (NUMR(N),N=1,NUMER)
INPT167      GO TO 412
INPT168 411 WRITE(6,2013)INCASF,NPMAX,NELMAX,NUMPC,NPMIS,MTYPE
INPT169 412 IF(NPMIS.FQ.0) GO TO 420
INPT170      WRITE(6,2014) (NNP(M),M=1,NPMIS)
INPT171 420 CONTINUE
INPT172      IF (NUMBER1.EQ.0) GO TO 440
INPT173      WRITE (6,2018) (NUMR1(N),N=1,NUMBER1)
INPT174      WRITE (6,2020) MTYPE1
INPT175 440 IF (KMORF.NF.0) GO TO 510
INPT176      WRITE (6,2025) MBAND
INPT177 C
INPT178 C      READ AND PRINT BOUNDARY PRESSURES, IF ANY
INPT179 C
INPT180      IF(NUMPC.FQ.0) GO TO 510
INPT181      WRITE (6,2006)
INPT182      DO 500 L=1,NUMPC
INPT183      READ(5,1005) IRC(L),JBC(L),PR(L,1),PR(L,2)
INPT184 500 WRITE(6,2009)IRC(L),JBC(L),PR(L,1),PR(L,2)
INPT185 510 CONTINUE
INPT186 C
INPT187 C      FOR ELEMENTS REMOVED OR NEWLY ADDED IN THIS STEP
INPT188 C      SET INITIAL STRESSES EQUAL TO ZERO
INPT189 C
INPT190      IF (NUMBER.FQ.0) GO TO 601
INPT191      DO 600 I=1,NUMBER
INPT192      NUM = NUMR(I)
INPT193      IX(NUM,5)=MTYPE
INPT194      SIGI(NUM,1)=0.0
INPT195      SIGI(NUM,2)=0.0
INPT196      SIGI(NUM,3)=0.0
INPT197      SIGI(NUM,4)=0.0
INPT198      MTAC(NUM)=0
INPT199 600 CONTINUE
INPT200 601 IF (NUMFPI.FQ.0) GO TO 610

```

```

INPT201      DD 606 I=1,NUMFR1
INPT202      NUM=NUMR1(I)
INPT203      606 IX(NUM,5)=MYPE1
INPT204      610 IF (KMURF.NF.0) GO TO 300
INPT205 C
INPT206 C      EVERY ELEMENT IS ASSUMED ELASTIC AT THE START OF EACH STEP
INPT207 C
INPT208      SR1=1.0
INPT209      SR2=0.0
INPT210      NPP=NP
INPT211      IF (NCASE.LT.1) NPP=1
INPT212      CALL SOLVE(P,Z,UP,UZ,CODE,BO,E,RO,PR,SIG1,TH,FR,AA(NI3),AA(NI4),
INPT213      *      IX,MTAG,IBC,JRC,NUMR,NUMR1,NNP)
INPT214      )IF(NCASE.LF.NSTEP) GO TO 300
INPT215      1001 FORMAT (115,1F10.0)
INPT216      1002 FORMAT (6F10.0)
INPT217      1006 FORMAT(A6)
INPT218      1003 FORMAT (15,F5.0,4F10.0)
INPT219      1004 FORMAT(615,5F10.0)
INPT220      1005 FORMAT (215,2F10.0)
INPT221      1007 FOPMAT(1615)
INPT222      1008 FORMAT(18A4)
INPT223      1009 FORMAT(1015,F10.0)
INPT224      2001 FOPMAT (17HOMATERIAL NUMFR= 13, 15H, MASS DENSITY= F12.4)
INPT225      2002 FORMAT(16HOFELASTIC MODULUS 14X 2HNU 8X 8HCOHESION 2X 14HFRICITION A
INPT226      .NGLF 2X 13HARFA FOR ONED /(2F16.5,3F16.5))
INPT227      2003 FORMAT (111H1NODAL POINT      TYPE X ORDNATE Y ORDNATE X LO
INPT228      .AO OR DISPLACEMENT Y LOAD OR DISPLACEMENT PORE PRESSURE )
INPT229      2004 FORMAT (112,F12.2,2F12.3,2F24.7)
INPT230      2005 FORMAT (26HONODAL POINT CARD ERROR N= 15)
INPT231      2006 FORMAT(109HIELEMENT NO.      I      J      K      L      MATERIAL      X-S
INPT232      .TRFSS      Y-STRESS      XY-STRESS      Z-STRESS      THICKNESS      )
INPT233      2007 FORMAT(112,416,1112,5F12.3)
INPT234      2008 FORMAT (29HOPPRESSURE BOUNDARY CONDITIONS/ 40H      I      J      PRESS
INPT235      .URE I      PRESSURE J      )
INPT236      2009 FORMAT (216,2F12.3)
INPT237      2011 FORMAT(1H1 18A4////)
INPT238      2012 FORMAT( 51H0 RESULTS AFTER STEP NO.----- 15/
INPT239      . 58H0 TOTAL NUMBER OF NODAL POINTS IN SYSTEM AT THIS STEP---- 15/
INPT240      . 58H0 TOTAL NUMBER OF ELEMENTS IN SYSTEM AT THIS STEP----- 15/
INPT241      . 58H0 TOTAL NUMBER OF PRESSURE CARDS AT THIS STEP----- 15/
INPT242      . 58H0 TOTAL NUMBER OF NODAL POINTS MISSING AT THIS STEP----- 15/
INPT243      . 58H0 MATERIAL TYPE OF ELEMNTS ADDED/REMOVED IN THIS STEP--- 15/
INPT244      . 58H0 ELEMENTS REMOVED IN THIS STEP ARE ----- /
INPT245      . (2015))
INPT246      2013 FOPMAT( 51H0 RESULTS AFTER STEP NO.----- 15/
INPT247      . 58H0 TOTAL NUMBER OF NODAL POINTS IN SYSTEM AT THIS STEP---- 15/
INPT248      . 58H0 TOTAL NUMBER OF ELEMENTS IN SYSTEM AT THIS STEP----- 15/
INPT249      . 58H0 TOTAL NUMBER OF PRESSURE CARDS AT THIS STEP----- 15/
INPT250      . 58H0 TOTAL NUMBER OF NODAL POINTS MISSING AT THIS STEP----- 15/

```

```

INPT251      . 58HO MATERIAL TYPE BEING PUT IN THIS STEP ----- 15/
INPT252      . 58HO ELEMENTS ADDED IN THIS STEP ARE ----- /
INPT253      . (20151)
INPT254 2014 FORMAT( 53HO NODAL POINTS REMOVED OR ADDED IN THIS STEP ARE-----
INPT255      ./(20151)
INPT256 2016 FORMAT(1R,2F8.2,1P4E12.4,OP1F8.5)
INPT257 2017 FOPMAT(110,2F20.7)
INPT258 2018 FOPMAT (52HO ELEMENTS FOR WHICH MATERIAL HAS BEEN CHANGED ARE--
INPT259      ./(20151)
INPT260 2020 FOPMAT (54HO NEW MATERIAL TYPE OF THE ABOVE CHANGED ELEMENTS IS--
INPT261      . 151
INPT262 2025 FORMAT(27H BAND WIDTH FOR THIS STEP 15)
INPT263 3000 FORMAT(//12H BAND WIDTH 15)
INPT264 4010 FORMAT( 65H THE MINIMUM DIMENSION OF AA AND NTOT NEEDED FOR THIS P
INPT265      .ROPLEM IS 110)
INPT266      RETURN
INPT267      END

```



```

SOLV 1      SUBROUTINE SOLVE (P,Z,UR,UZ,CONF,RO,E,RO,PR,SIGI,TH,FR,B,A,IX,
SOLV 2      *      MTAG,IRC,JRC,NUMR,NUMR1,NNP)
SOLV 3      IMPLICIT REAL*8(A-H,O-Z)
SOLV 4      COMMON/TOTAL/ AA(30000), JA(7000)
SOLV 5      COMMON/ONE/NUMNP,NUMEL,NUMMAT,NUMPC,NPMAX,NFLMAX,N,NNN,KKK,KLK,
SOLV 6      .      ISHIFT,MBAND,NBAND,NUMBLK,MTYPE,NCASE,MCASE,NMR,NL,NPP,NFQ,NPC
SOLV 7      .      NP,NSTEP,NCODE
SOLV 8      COMMON/TWO/RR(5),ZZ(5),S(10,10),P(10),ST(3,10),C(4,4),SIG(7),FF(4)
SOLV 9      .      ,TITLE(18),SR1,SR2,XC,YC,ACFLR,ACFLZ,BCFLR,BCFLZ,VOL
SOLV 10     COMMON/THREE/STOP,MTOT,NTOT,N13,N14,MR
SOLV 11     DIMENSION P(NUMNP),Z(NUMNP),UR(NUMNP),UZ(NUMNP),CODE(NUMNP),
SOLV 12     .      RO(NEL),E(5,NUMMAT),RO(NUMMAT),PR(NPC,2),SIGI(NUMFL,8),
SOLV 13     .      TH(NUMEL),FR(NUMEL),IX(NUMFL,5),MTAG(NUMEL),IRC(NPC),
SOLV 14     .      JBC(NPC),NUMR(NMR),NUMR1(NMR),NNP(NMR)
SOLV 15     DIMENSION B(NL),A(NL,NBAND)
SOLV 16     DIMENSION LM(4)
SOLV 17 C
SOLV 18     DO 750 NNN=1,NPP
SOLV 19     KKK=0
SOLV 20     REWIND 2
SOLV 21     ND=NL/2
SOLV 22     NR=ND/2
SOLV 23     STOP=0.0
SOLV 24     NUMBLK=0
SOLV 25     DO 50 N=1,NL
SOLV 26     B(N)=0.0
SOLV 27     DO 50 M=1,MBAND
SOLV 28     50 A(N,M)=0.0
SOLV 29     IF(NCASE.GT.MCASE) GO TO 51
SOLV 30     IF(NNN.GT.1) GO TO 51
SOLV 31     BCFLR =ACFLR
SOLV 32     BCFLZ =ACFLZ
SOLV 33     51 CONTINUE
SOLV 34     IF(NNN.GT.1) GO TO 52
SOLV 35     ACFLR =BCFLR
SOLV 36     ACFLZ =BCFLZ
SOLV 37     GO TO 55
SOLV 38     52 ACFLR =0.0
SOLV 39     ACFLZ =0.0
SOLV 40     NUMPC =0
SOLV 41     55 CONTINUE
SOLV 42     60 NUMBLK=NUMBLK+1
SOLV 43     NH=NR*(NUMBLK+1)
SOLV 44     NM=NH-NR
SOLV 45     NS=NM-NB+1
SOLV 46     KSHIFT=2*NS-2
SOLV 47     DO 210 N=1,NELMAX
SOLV 48     IF(IX(N,5).LE.0) GO TO 210
SOLV 49     MTYPE=IX(N,5)
SOLV 50     IF (RO(MTYPE).EQ.0.) GO TO 210

```

```

SOLV 51      DO 80 I=1,4
SOLV 52      IF (IX(N,1).LT.NSI) GO TO 80
SOLV 53      IF (IX(N,1).LT.NM) GO TO 90
SOLV 54      80 CONTINUE
SOLV 55      GO TO 210
SOLV 56      90 IF (IX(N,2).NE.IX(N,21)) GO TO 95
SOLV 57      CALL ONED (P,Z,UP,UZ,CONF,RQ,F,RQ,PR,SIG1,TH,FP,IX,MTAG,
SOLV 58      *      IRC,JRC,NUMP,NUMP1,NNP)
SOLV 59      IX(N,5)=-IX(N,5)
SOLV 60      MM=2
SOLV 61      GO TO 130
SOLV 62      95 CALL QUAD (P,Z,UR,UZ,CONF,RQ,E,RQ,PR,SIG1,TH,FR,IX,MTAG,
SOLV 63      *      IRC,JRC,NUMP,NUMP1,NNP)
SOLV 64      IX(N,5)=-IX(N,5)
SOLV 65      IF (VOL.GT.0.1) GO TO 110
SOLV 66      WRITE(6,2000) N
SOLV 67      STOP=1.0
SOLV 68      110 MM=4
SOLV 69      IF (IX(N,31).EQ.IX(N,4)) MM=3
SOLV 70      130 DO 140 I=1,MM
SOLV 71      140 LM(I)=2*IX(N,11-2
SOLV 72      DO 200 I=1,MM
SOLV 73      DO 200 K=1,2
SOLV 74      II=LM(I)+K-KSHIFT
SOLV 75      KK=2*I-2+K
SOLV 76      R(II)=B(II)+P(KK)
SOLV 77      DO 200 J=1,MM
SOLV 78      DO 200 L=1,2
SOLV 79      JJ=LM(J)+L-11+1-KSHIFT
SOLV 80      LL=2*J-2+L
SOLV 81      IF (JJ.LE.0) GO TO 200
SOLV 82      A(II,JJI)=A(II,JJI)+S(KK,LL)*TH(N)
SOLV 83      200 CONTINUE
SOLV 84      210 CONTINUE
SOLV 85      DO 220 N=NS,NM
SOLV 86      IF (N.GT.NPMAX) GO TO 220
SOLV 87      K=2*N-KSHIFT
SOLV 88      B(K)=B(K)+UZ(N)
SOLV 89      R(K-1)=R(K-1)+UR(N)
SOLV 90      220 CONTINUE
SOLV 91      C
SOLV 92      IF (NUMPC.EQ.0) GO TO 310
SOLV 93      DO 300 L=1,NUMPC
SOLV 94      I=IRC(L)
SOLV 95      J=JRC(L)
SOLV 96      DR=Z(11)-Z(J)
SOLV 97      DZ=P(J)-P(I)
SOLV 98      PP2=(PR(L,2)+PR(L,1))/6.
SOLV 99      PP1=PP2+PR(L,1)/6.
SOLV 100     PP2=PP2+PR(L,2)/6.

```

```

SOLV101      II=2*I-KSHIFT
SOLV102      JJ=2*J-KSHIFT
SOLV103      IF((II.LE.0).OR.(II.GT.ND)) GO TO 265
SOLV104      B(II-1)=B(II-1)+PP1*DR
SOLV105      R(II)=R(II)+PP1*DZ
SOLV106      265 IF((JJ.LE.0).OR.(JJ.GT.ND)) GO TO 300
SOLV107      R(JJ-1)=R(JJ-1)+PP2*DR
SOLV108      B(JJ)=B(JJ)+PP2*DZ
SOLV109      300 CONTINUE
SOLV110      310 DO 400 M=NS,NH
SOLV111          IF(M.GT.NPMAX) GO TO 400
SOLV112          U=UR(M)
SOLV113          N=2*M-1-KSHIFT
SOLV114          IF (CONF(M)) 390,400,316
SOLV115      316 IF (CODE(M).EQ.1.) GO TO 370
SOLV116          IF (CONF(M).EQ.2.) GO TO 390
SOLV117          IF (CONF(M)-3.) 390,380,390
SOLV118      370 CALL MODIFY(A,R,NL ,MBAND,NRAND,N,U)
SOLV119          GO TO 400
SOLV120      380 CALL MODIFY(A,R,NL ,MBAND,NRAND,N,U)
SOLV121      390 U=UZ(M)
SOLV122          N=N+1
SOLV123          CALL MODIFY(A,R,NL ,MBAND,NRAND,N,U)
SOLV124      400 CONTINUE
SOLV125          WRITE (2) (P(N),(A(N,M),M=1,MBAND),N=1,ND)
SOLV126          DO 420 N=1,ND
SOLV127              K=N+ND
SOLV128              B(N)=B(K)
SOLV129              R(K)=0.0
SOLV130              DO 420 M=1,MBAND
SOLV131                  A(N,M)=A(K,M)
SOLV132      420 A(K,M)=0.0
SOLV133          IF(NM.LT.NPMAX) GO TO 60
SOLV134          IF (STOP.NE.0.) CALL EXIT
SOLV135          DO 600 N=1,NUMNP
SOLV136              UZ(N)=0.
SOLV137      600 UR(N)=0.
SOLV138              N15=N14+2*NUMNP
SOLV139              N16=N15+4*NUMFL
SOLV140              N17=N16+NUMFL
SOLV141              CALL RANSOL (N1,NRAND,MBAND,NUMFLK,B,A)
SOLV142              CALL STRESS (R,Z,UR,UZ,CONF,BD,F,PD,PR,SIGI,TH,FP,B,AA(N15),
SOLV143              *      AA(N16),AA(N17),IX,MTAG,IRC,JRC,NUMR,NUMR1,NNP)
SOLV144              DO 700 N=1,NPMAX
SOLV145                  NN=2*N
SOLV146                  BO(NN-1)=BO(NN-1)+R(NN-1)
SOLV147      700 BO(NN)=BO(NN)+B(NN)
SOLV148 C
SOLV149 C      PRINT OUT CUMULATIVE DISPLACEMENTS
SOLV150 C

```

```

SOLV151      WRITE(6,2002)(N,BO(2*N-1),BO(2*N),N=1,NUMNP)
SOLV152      IF(KLK.EQ.1) GO TO 750
SOLV153      IF((KKK.EQ.0).OR.(SR2.GT.1.)) GO TO 800
SOLV154      750 CONTINUE
SOLV155      800 IF(NUMER.GT.0) GO TO 850
SOLV156      IF(NCASE.GT.0) GO TO 850
SOLV157 C
SOLV158 C      IF INITIAL STRESSES ARE EVALUATED IN THIS STEP INITIALIZE BO
SOLV159 C
SOLV160      DO 810 N=1,NEQ
SOLV161      810 BO(N)=0.
SOLV162      850 NCASE=NCASE+1
SOLV163      DO 900 N=1,NFLMAX
SOLV164      900 IF(FR(N).GT.1.) FR(N)=1.
SOLV165      RETURN
SOLV166      1000 FORMAT (1H1,40X,'LOAD MATRIX'/(10E13.4))
SOLV167      2000 FORMAT (26H0NEGATIVE AREA ELEMENT NO. 14)
SOLV168      2002 FORMAT(12H1 N.P NUMRER 1RX 2HUX 1RX 2HUY /(112,2F20.7))
SOLV169      END

```

```

0NFD 1      SUBROUTINE 0NFD (R,Z,UP,UZ,CODE,BO,F,RO,PO,SIG1,TH,FR,IX,MTAG,
0NFD 2      *      IRC,JBC,NUMR,NUMR1,NNP)
0NFD 3      IMPLICIT REAL*8 (A-H,O-Z)
0NFD 4      COMMON/TOTAL/ AA(30000), IA(7000)
0NFD 5      COMMON/0NF/NUMNP,NUMFL,NUMMAT,NUMPC,NPMAX,NFLMAX,N,NNN,KKK,KLK,
0NFD 6      .      ISHIFT,MRAND,NRAND,NUMBLK,MTYPE,NCASE,MCASE,NMR,NL,NPP,NEQ,NPC
0NFD 7      .      NP,NSTEP,NCODE
0NFD 8      COMMON/TWO/RR(5),ZZ(5),S(10,10),P(10),ST(3,10),C(4,4),SIG(7),EE(4)
0NFD 9      .      TITLE(18),SR1,SR2,XC,YC,ACFLR,ACELZ,BCFLR,BCFLZ,VOL
0NFD 10     COMMON/THREF/STOP,MTOT,NTOT,N13,N14,MR
0NFD 11     DIMENSION R(NUMNP),Z(NUMNP),UP(NUMNP),UZ(NUMNP),CODE(NUMNP),
0NFD 12     .      BO(NFQ),F(5,NUMMAT),RO(NUMMAT),PR(NPC,2),SIG1(NUMEL,8),
0NFD 13     .      TH(NUMEL),FR(NUMFL),IX(NUMFL,5),MTAG(NUMFL),IRC(NPC),
0NFD 14     .      JBC(NPC),NUMR(NMR),NUMR1(NMR),NNP(NMR)
0NFD 15 C
0NFD 16     DO 100 I=1,4
0NFD 17     P(I)=0.0
0NFD 18     DO 100 J=1,8
0NFD 19     100 S(I,J)=0.0
0NFD 20     MTYPE=IX(N,5)
0NFD 21     I=IX(N,1)
0NFD 22     J=IX(N,2)
0NFD 23     DX=P(J)-R(I)
0NFD 24     DY=Z(J)-Z(I)
0NFD 25     XL=DSORT(DX**2+DY**2)
0NFD 26     COSA=DX/XL
0NFD 27     SINA=DY/XL
0NFD 28     COMM= F(1,MTYPE)*F(5,MTYPE)/XL
0NFD 29     S(1,1)=COSA*COSA*COMM
0NFD 30     S(1,2)=COSA*SINA*COMM
0NFD 31     S(1,3)=-S(1,1)
0NFD 32     S(1,4)=-S(1,2)
0NFD 33     S(2,1)=S(1,2)
0NFD 34     S(2,2)=SINA*SINA*COMM
0NFD 35     S(2,3)=-S(1,2)
0NFD 36     S(2,4)=-S(2,2)
0NFD 37     S(3,1)=S(1,3)
0NFD 38     S(3,2)=S(2,3)
0NFD 39     S(3,3)=S(1,1)
0NFD 40     S(3,4)=S(1,2)
0NFD 41     S(4,1)=S(1,4)
0NFD 42     S(4,2)=S(2,4)
0NFD 43     S(4,3)=S(3,4)
0NFD 44     S(4,4)=S(2,2)
0NFD 45     II=4
0NFD 46     IF(NNN.FQ.1) II=0
0NFD 47     FP=SIG1(N,II+1)/F(1,MTYPE)
0NFD 48     DX=DX*FP
0NFD 49     DY=DY*FP
0NFD 50     P(1)=S(1,1)*DX+S(1,2)*DY

```

```
ONED 51      P(2)=S(2,1)*DX+S(2,2)*DY
ONED 52      P(3)=-P(1)
ONED 53      P(4)=-P(2)
ONED 54      RETURN
ONED 55      END
```



```

QUAD 1  SUBROUTINE QUAD (R,Z,UR,UZ,CODE,BD,E,RO,PR,SIG1,TH,FR,IX,MTAG,
QUAD 2  * IBC,JBC,NUMP,NUMR1,NNP)
QUAD 3  IMPLICIT REAL*8(A-H,O-Z)
QUAD 4  COMMON/TOTAL/ AA(30000), IA(7000)
QUAD 5  COMMON/DNF/NUMNP,NUMFL,NUMMAT,NUMPC,NPMAX,NFLMAX,N,NNN,KKK,KLK,
QUAD 6  * ISHIFT,MRAND,NBAND,NUMBLK,MTYPE,NCASE,MCASE,NMR,NL,NPP,NEQ,NPC
QUAD 7  * NP,NSTFP,NCODE
QUAD 8  COMMON/TWO/PP(5),ZZ(5),S(10,10),P(10),ST(3,10),C(4,4),SIG(7),FF(4)
QUAD 9  * ,TITLE(18),SR1,CR2,XC,YC,ACELR,ACFLZ,BCFLR,BCFLZ,VOL
QUAD 10 COMMON/THREF/STOP,MTOT,NTOT,N13,N14,MR
QUAD 11 DIMENSION R(NUMNP),Z(NUMNP),UR(NUMNP),UZ(NUMNP),CODE(NUMNP),
QUAD 12 * BO(NFQ),E(5,NUMMAT),PD(NUMMAT),PR(NPC,2),SIG1(NUMEL,8),
QUAD 13 * TH(NUMFL),FR(NUMFL),IX(NUMFL,5),MTAG(NUMEL),IBC(NPC),
QUAD 14 * JBC(NPC),NUMR(NMR),NUMR1(NMR),NNP(NMR)
QUAD 15 DIMENSION LM(4),U(3),V(3)
QUAD 16 C
QUAD 17 CALL STPSTR (R,Z,UP,UZ,CODE,RO,E,RO,PR,SIG1,TH,FR,IX,MTAG,
QUAD 18 * IBC,JBC,NUMR,NUMP1,NNP)
QUAD 19 DO 130 J=1,10
QUAD 20 P(J)=0.
QUAD 21 DO 120 I=1,3
QUAD 22 120 ST(I,J)=0.
QUAD 23 DO 130 I=1,10
QUAD 24 130 S(I,J)=0.
QUAD 25 DO 140 I=1,4
QUAD 26 NPN=IX(N,1)
QUAD 27 RR(1)=R(NPN)
QUAD 28 140 ZZ(1)=Z(NPN)
QUAD 29 XC=(RR(1)+PR(2)+RR(3)+PR(4))/4.
QUAD 30 YC=(ZZ(1)+ZZ(2)+ZZ(3)+ZZ(4))/4.
QUAD 31 RR(5)=XC
QUAD 32 ZZ(5)=YC
QUAD 33 K=5
QUAD 34 J=1
QUAD 35 I=4
QUAD 36 LM(3)=0
QUAD 37 NT=4
QUAD 38 IF(IX(N,3).NE.IX(N,4)) GO TO 160
QUAD 39 NT=1
QUAD 40 LM(3)=5
QUAD 41 I=1
QUAD 42 K=3
QUAD 43 J=2
QUAD 44 XC=(RR(1)+RR(2)+RR(3))/3.
QUAD 45 YC=(ZZ(1)+ZZ(2)+ZZ(3))/3.
QUAD 46 RR(5)=PR(3)
QUAD 47 ZZ(5)=ZZ(3)
QUAD 48 160 DO 200 NN=1,NT
QUAD 49 LM(1)=2*I-1
QUAD 50 LM(2)=2*J-1

```

```

QUAD 51      U(1)=ZZ(J1-ZZ(K)
QUAD 52      U(2)=ZZ(K)-ZZ(I)
QUAD 53      U(3)=ZZ(I)-ZZ(J)
QUAD 54      V(1)=RR(K)-RR(J)
QUAD 55      V(2)=RR(I)-RR(K)
QUAD 56      V(3)=RR(J)-RR(I)
QUAD 57      ARFA=(RP(J)*U(2)+RR(I)*U(1)+RR(5)*U(3))/2.
QUAD 58      VOL=VOL+AREA*TH(N)
QUAD 59      COMM=.25/ARFA
QUAD 60      XNT=NT
QUAD 61      COM=2.0/XNT
QUAD 62      COM=COM*COMM
QUAD 63      DX=ARFA*TH(N)*RD(MTYPE)/3.
QUAD 64      DY=DX*ACFLZ
QUAD 65      DX=DX*ACFLR
QUAD 66      DO 180 L=1,3
QUAD 67      II=LM(L)
QUAD 68      ST(1,II)=ST(1,II)+U(L)*COM
QUAD 69      ST(2,II+1)=ST(2,II+1)+V(L)*COM
QUAD 70      ST(3,II)=ST(3,II)+V(L)*COM
QUAD 71      ST(3,II+1)=ST(3,II+1)+U(L)*COM
QUAD 72      P(II)=P(II)+DX
QUAD 73      P(II+1)=P(II+1)+DY
QUAD 74      DO 180 M=1,3
QUAD 75      JJ=LM(M)
QUAD 76      S(II,JJ)=S(II,JJ)+(U(L)*C(1,1)*U(M)+V(L)*C(2,3)*V(M)+V(L)*C(1,3)*U
QUAD 77      .(M)+U(L)*C(1,3)*V(M))*COMM
QUAD 78      S(II,JJ+1)=S(II,JJ+1)+(U(L)*C(1,2)*V(M)+V(L)*C(3,3)*U(M)+V(L)*C(2,
QUAD 79      .3)*V(M)+U(L)*C(1,3)*U(M))*COMM
QUAD 80      S(II+1,JJ+1)=S(II+1,JJ+1)+(V(L)*C(2,2)*V(M)+U(L)*C(3,3)*U(M)+U(L)*
QUAD 81      .C(2,3)*V(M)+V(L)*C(2,3)*U(M))*COMM
QUAD 82      S(JJ+1,II)=S(II,JJ+1)
QUAD 83      180 CONTINUE
QUAD 84      I=J
QUAD 85      J=J+1
QUAD 86      200 CONTINUE
QUAD 87      IF(IX(N,3).EQ.IX(N,4)) GO TO 250
QUAD 88      DO 240 I=1,2
QUAD 89      KK=IN-I
QUAD 90      DO 240 K=1,KK
QUAD 91      CC=S(KK+1,K)/S(KK+1,KK+1)
QUAD 92      P(K)=P(K)-CC*P(KK+1)
QUAD 93      DO 230 J=1,3
QUAD 94      230 ST(J,K)=ST(J,K)-CC*ST(J,KK+1)
QUAD 95      DO 240 J=1,KK
QUAD 96      240 S(J,K)=S(J,K)-CC*S(J,KK+1)
QUAD 97      250 CONTINUE
QUAD 98      II=0
QUAD 99      IF (NNN.GT.1) II=4
QUAD 100     SIG(1)=-SIG(N,II+1)

```

```

QUAD101      SIG(2)=-SIG1(N,11+2)
QUAD102      SIG(3)=-SIG1(N,11+3)
QUAD103      DO 520 I=1,8
QUAD104      DO 510 J=1,3
QUAD105      510 P11=P11+STIJ,1)*SIG1J)*VOL
QUAD106      520 CONTINUE
QUAD107      RETURN
QUAD108      END

```

```

STST 1      SUPROUTINE STRSTR (P,Z,UP,UZ,CODF,BD,F,RO,PR,SIG),TH,FP,IX,MTAG,
STST 2      *      )BC,JBC,NUMR,NUMR1,NNP)
STST 3      )MPL)C)T RFAL*R(A-H,O-Z)
STST 4      COMMON/TOTAL/ AA(30000), IA(7000)
STST 5      COMMON/ONE/NUMNP,NUMEL,NUMMAT,NUMPC,NPMAX,NELMAX,N,NNN,KKK,KLK,
STST 6      .      )SHIFT,MRAND,NBAND,NUMBLK,MTYPE,NCASE,MCASE,NMR,NL,NPP,NFQ,NPC
STST 7      .      ,NP,NSTEP,NCODE
STST 8      COMMON/TWO/RR(5),ZZ(5),S(10,10),P(10),ST(3,10),C(4,4),SIG(7),EF(4)
STST 9      .      ,T)TLF(18),SR1,SR2,XC,YC,ACELR,ACELZ,RCELR,RCELZ,VOL
STST 10     COMMON/THREE/STOP,MTOT,NTOT,N13,N14,MR
STST 11     DIMENSION R(NUMNP),Z(NUMNP),UP(NUMNP),UZ(NUMNP),CODE(NUMNP),
STST 12     .      BD(NEQ),E(5,NUMMAT),RO(NUMMAT),PR(NPC ,2),SIG(NUMFL,8),
STST 13     .      TH(NUMEL),FR(NUMEL),IX(NUMEL,5),MTAG(NUMEL),)PC(NPC),
STST 14     .      JRC(NPC ),NUMR(NMR),NUMR1(NMR),NNP(NMP)
STST 15 C
STST 16     MTYPE=X(N,5)
STST 17     VOL=0.
STST 18     DO 50 KK=1,4
STST 19     50 FE(KK)=F(KK,MTYPE)
STST 20     )F((NCASE.GT.1).AND.(NNN.EQ.1)) GO TO 60
STST 21     IF(MTAG(N).GT.0) GO TO 70
STST 22     60 CC=FE(2)/(1.-FE(2))
STST 23     BP=FE(1)/(1.-FE(2)**2)
STST 24     COMM=BP/(1.-CC**2)
STST 25     C(1,1)=COMM
STST 26     C(1,2)=COMM*CC
STST 27     C(1,3)=0.
STST 28     C(1,4)=0.
STST 29     C(2,1)=C(1,2)
STST 30     C(2,2)=C(1,1)
STST 31     C(2,3)=0.
STST 32     C(2,4)=0.
STST 33     C(3,1)=0.
STST 34     C(3,2)=0.
STST 35     C(3,3)=.5*COMM*(1.-CC)
STST 36     C(3,4)= 0.
STST 37     C(4,1)= C(1,2)
STST 38     C(4,2)= C(1,2)
STST 39     C(4,3)= 0.
STST 40     C(4,4)= 0.
STST 41     CC= 2.*DSIN(FF(4)/57.296)
STST 42     RR= 1.732*(3.-DSIN(EE(4)/57.296))
STST 43     PP=6.*DCOS(FF(4)/57.296)
STST 44     EE(4)=CC/RR
STST 45     FF(3)= FF(3)*PP/RR
STST 46     GO TO 500
STST 47     70 CC= 2.*DSIN(FF(4)/57.296)
STST 48     RR= 1.732*(3.-DSIN(EE(4)/57.296))
STST 49     PP=6.*DCOS(FF(4)/57.296)
STST 50     EE(4)=CC/PP

```

```

STST 51      FF(2)= FF(3)*PP/BB
STST 52      CC=2.*(1.+FF(2))/(3.-6.*FF(2))
STST 53      DD=SIGI(N,1)-SIGI(N,2)
STST 54      FF=SIGI(N,1)-SIGI(N,4)
STST 55      GG=SIGI(N,2)-SIGI(N,4)
STST 56      BJ2= (DD**2+FF**2+GG**2)/6. +SIGI(N,3)**2
STST 57      BJ2=DSQRT(BJ2)
STST 58      BJ1=SIGI(N,1)+SIGI(N,2)+SIGI(N,4)
STST 59      DD=BJ1/BJ2
STST 60      BB=1.+9.*(FF(4)**2)*CC
STST 61      CC=3.*FF(4)*CC-DD/3.
STST 62      DD=FF(4)-DD/6.
STST 63      H1=.5*CC/(BB*BJ2)
STST 64      H2=DD*CC/BB-FF(2)*FF(3)/(BB*BJ2*(1.-7.*FF(2)))
STST 65      H3=.5/(BB*BJ2*BJ2)
STST 66      BB=FF(1)/(1.+FF(2))
STST 67      C(1,1)=BB*(1.-H2-2.*H1*SIGI(N,1)-H3*(SIGI(N,1)**2))
STST 68      C(1,2)=-BB*(H2+H1*(SIGI(N,1)+SIGI(N,2))+H3*SIGI(N,1)*SIGI(N,2))
STST 69      C(1,3)=-BB*(H1*SIGI(N,3)+H3*SIGI(N,1)*SIGI(N,3))
STST 70      C(1,4)=0.
STST 71      C(2,1)=C(1,2)
STST 72      C(2,2)=BB*(1.-H2-2.*H1*SIGI(N,2)-H3*(SIGI(N,2)**2))
STST 73      C(2,3)=-BB*(H1*SIGI(N,3)+H3*SIGI(N,2)*SIGI(N,3))
STST 74      C(2,4)=0.
STST 75      C(3,1)=C(1,3)
STST 76      C(3,2)=C(2,3)
STST 77      C(3,3)=BB*(.5-H3*(SIGI(N,3)**2))
STST 78      C(3,4)=0.
STST 79      C(4,1)=-BB*(H2+H1*SIGI(N,1)+H1*SIGI(N,4)+H3*SIGI(N,1)*SIGI(N,4))
STST 80      C(4,2)=-BB*(H2+H1*SIGI(N,2)+H1*SIGI(N,4)+H3*SIGI(N,2)*SIGI(N,4))
STST 81      C(4,3)=-BB*(H1*SIGI(N,3)+H3*SIGI(N,3)*SIGI(N,4))
STST 82      C(4,4)=0.
STST 83      500 RETURN
STST 84      END

```

```

MODI 1      SUBROUTINE MODIFY(A,B,NL ,MBAND,NBAND,N,U)
MODI 2      IMPLICIT REAL*8(A-H ,O-Z)
MODI 3      DIMENSION B(NL),A(NL,NBAND)
MODI 4 C
MODI 5      DO 250 M=2,MBAND
MODI 6          K=N-M+1
MODI 7          IF(K.LE.0) GO TO 235
MODI 8          B(K)=B(K)-A(K,M)*U
MODI 9          A(K,M)=0.0
MODI 10      235 K=N+M-1
MODI 11          IF(NL .LT.K) GO TO 250
MODI 12          B(K)=B(K)-A(N,M)*U
MODI 13          A(N,M)=0.0
MODI 14      250 CONTINUE
MODI 15          A(N,1)=1.0
MODI 16          P(N)=U
MODI 17          RETURN
MODI 18      END

```



```

RANS 1      SUBROUTINE BANSOL (ND,NBAND,MM,NUMBLK,B,A)
RANS 2      IMPLICIT REAL*8(A-H,O-Z)
RANS 3      DIMENSION B(ND),A(ND,NBAND)
RANS 4 C
RANS 5      NN=ND/?
RANS 6      NL=NN+1
RANS 7      NH=NN+NN
RANS 8      REWIND 1
RANS 9      REWIND 2
RANS 10     NR=0
RANS 11     GO TO 150
RANS 12     100 NB=NR+1
RANS 13     DO 125 N=1,NN
RANS 14     NM=NN+N
RANS 15     B(N)=B(NM)
RANS 16     P(NM)=0.0
RANS 17     DO 125 M=1,MM
RANS 18     A(N,M)=A(NM,M)
RANS 19     125 A(NM,M)=0.0
RANS 20     IF(NUMBLK.EQ.NR) GO TO 200
RANS 21     150 READ (2) (B(N),(A(N,M),M=1,MM),N=NL,NH)
RANS 22     IF(NB.EQ.0) GO TO 100
RANS 23     200 DO 300 N=1,NN
RANS 24     IF(A(N,1).EQ.0.) GO TO 300
RANS 25     R(N)=B(N)/A(N,1)
RANS 26     DO 275 L=2,MM
RANS 27     IF (A(N,L).EQ.0.) GO TO 275
RANS 28     C=A(N,L)/A(N,1)
RANS 29     I=N+L-1
RANS 30     J=0
RANS 31     DO 250 K=L,MM
RANS 32     J=J+1
RANS 33     250 A(I,J)=A(I,J)-C*A(N,K)
RANS 34     R(I)=B(I)-A(N,L)*P(N)
RANS 35     A(N,L)=C
RANS 36     275 CONTINUE
RANS 37     300 CONTINUE
RANS 38     IF(NUMBLK.EQ.NR) GO TO 400
RANS 39     WRITE (1) (R(N),(A(N,M),M=2,MM),N=1,NN)
RANS 40     GO TO 100
RANS 41     400 DO 450 M=1,NN
RANS 42     N=NN+1-M
RANS 43     DO 425 K=2,MM
RANS 44     L=N+K-1
RANS 45     425 R(N)=B(N)-A(N,K)*B(L)
RANS 46     NM=N+NN
RANS 47     B(NM)=B(N)
RANS 48     450 A(NM,NR)=B(N)
RANS 49     NB=NR+1
RANS 50     IF(NB.EQ.0) GO TO 500

```

```

FANS 51      RACKSPACE 1
FANS 52      RFAD (1) (B(N), (A(N,M), M=2,MM), N=1,NN)
FANS 53      RACKSPACE 1
FANS 54      GO TO 400
FANS 55      500 K=0
FANS 56      DO 600 N8=1,NUMBLK
FANS 57      DO 600 N=1,NN
FANS 58      K=K+1
FANS 59      NM=N+NN
FANS 60      600 B(K)=A(NM,NP)
FANS 61      RETURN
FANS 62      END

```

```

STRF 1      SUBROUTINE STRESS (R,Z,UR,UZ,COEF,BO,F,RO,PR,SIGI,TH,FR,B,EP,
STRF 2      *      RATIO,NEW,IX,MTAG,IBC,JBC,NUMR,NUMR1,NNP)
STRF 3      IMPLICIT REAL*8(A-H,O-Z)
STRF 4      COMMON/TOTAL/ AA(30000), IA(7000)
STRF 5      COMMON/ONE/NUMNP,NUMEL,NUMMAT,NUMPC,NPMAX,NELMAX,N,NNN,KKK,KLK,
STRF 6      .      ISHIFT,MBAND,NBAND,NUMBLK,MTYPE,NCASE,MCASE,NMR,NL,NPP,NEQ,NPC
STRF 7      .      NP,NSTEP,NCODE
STRF 8      COMMON/TWO/RR(5),ZZ(5),S(10,10),P(10),ST(3,10),C(4,4),SIG(7),FE(4)
STRF 9      .      ,TITLE(18),SR1,SR2,XC,YC,ACELR,ACELZ,BCELR,BCELZ,VOL
STRF 10     COMMON/THREE/STOP,MTOT,NTOT,N13,N14,M8
STRF 11     DIMENSION R(NUMNP),Z(NUMNP),UR(NUMNP),UZ(NUMNP),CODE(NUMNP),
STRF 12     .      BO(NEQ),E(5,NUMMAT),RO(NUMMAT),PR(NPC,2),SIGI(NUMEL,8),
STRF 13     .      TH(NUMEL),FR(NUMEL),IX(NUMEL,5),MTAG(NUMEL),IRC(NPC),
STRF 14     .      JBC(NPC),NUMR(NMR),NUMR1(NMR),NNP(NMR)
STRF 15     DIMENSION EP(NUMEL,4),RATIO(NUMEL),NEW(100),TT(4),TF(4)
STRF 16     DIMENSION B(NEQ)
STRF 17 C
STRF 18     TOLL=0.
STRF 19     SR=1.
STRF 20     SRATIO=1.0
STRF 21     NMY=0
STRF 22     MPRINT=0
STRF 23     KK=0
STRF 24     NOPT=0
STRF 25     TOLLA=10.
STRF 26     KJK=0
STRF 27     KKK=0
STRF 28     IF((NCASE.LE.1).OR.(NNN.GT.1)) GO TO 60
STRF 29     DO 50 N=1,NELMAX
STRF 30     IF (MTAG(N).GT.0) KJK=KJK+1
STRF 31     50 CONTINUE
STRF 32     60 CONTINUE
STRF 33     DO 300 N=1,NELMAX
STRF 34     RATIO(N)=1.
STRF 35     IX(N,5)=IABS(IX(N,5))
STRF 36     MTYPE=IX(N,5)
STRF 37     IF((RO(MTYPE).EQ.0.).OR.(IX(N,3).FQ.IX(N,2))) GO TO 300
STRF 38     CALL QUAD (R,Z,UR,UZ,COEF,BO,F,RO,PR,SIGI,TH,FR,IX,MTAG,
STRF 39     *      IBC,JBC,NUMR,NUMR1,NNP)
STRF 40     MM=4
STRF 41     IF(IX(N,3).FQ.IX(N,4)) MM=3
STRF 42     DO 175 I=1,4
STRF 43     EP(N,I) = 0.
STRF 44     DO 175 J=1,MM
STRF 45     II=2+J
STRF 46     JJ=2*IX(N,J)
STRF 47     175 EP(N,I)=EP(N,I)+ST(1,II)*B(JJ) + ST(1,II-1)*B(JJ-1)
STRF 48     DO 190 I=1,4
STRF 49     SIG(I)=0.0
STRF 50     DO 190 J=1,3

```

```

STPF 51 190 SIG((=SIG(1(+C(1,J)*FP(N,J)
STPF 52 DO 195 1=1,4
STPF 53 11=1+4
STPF 54 195 SIG(N,11)=SIG(N,1(+SIG(1(
STPF 55 DD=SIG(N,5(-SIG(N,6(
STPF 56 FF=SIG(N,6(-SIG(N,8(
STPF 57 GG=SIG(N,5(-SIG(N,8(
STPF 58 AJ2= (DD**2+GG**2+FF**2)/6.0 +SIG(N,7)**2
STPF 59 AJ2=DSORT(AJ2(
STPF 60 AJ1=SIG(N,5(+S(IG(N,6)+SIG(N,8)
STPF 61 FAIL=AJ2+FF(4(*AJ1
STPF 62 IF(MTAG(N(.EQ.0) GO TO 200
STPF 63 IF (FF(3(.EQ.0.( GO TO 300
STPF 64 IF ((NCASF.GT.1).AND.(NNN.FQ.1(( GO TO 198
STPF 65 DD=DAPS(FAIL-FF(3(
STPF 66 CHECK =0.05*FE(3(
STPF 67 IF(DD.LF.CHECK) GO TO 300
STPF 68 KKK=1
STPF 69 CR=CHECK/DD
STPF 70 IF(CR.GE.SR) GO TO 300
STPF 71 SR=CR
STPF 72 NOPT=1
STPF 73 JJI=N
STPF 74 GO TO 300
STPF 75 198 IF(FAIL.GT.FF(3( GO TO 300
STPF 76 MTAG(N(=0
STPF 77 KJK=KJK-1
STPF 78 GO TO 300
STPF 79 200 CONTINUE
STPF 80 IF(FAIL.LT.FF(3( GO TO 300
STPF 81 KK=KK+1
STPF 82 DD=SIG(N,1)-SIG(N,2)
STPF 83 FF=SIG(N,1)-SIG(N,4(
STPF 84 GG=SIG(N,2(-SIG(N,4(
STPF 85 AJ2= (DD**2+GG**2+FF**2)/6.0 +SIG(N,3)**2
STPF 86 AJ1=SIG(N,1(+SIG(N,2)+SIG(N,4)
STPF 87 DD=SIG(N,5)-SIG(N,6(
STPF 88 FF=SIG(N,6)-SIG(N,8(
STPF 89 GG=SIG(N,5(-SIG(N,8(
STPF 90 BJ2= (DD**2+GG**2+FF**2)/6.0 +SIG(N,7)**2
STPF 91 BJ1=SIG(N,5)+SIG(N,6)+SIG(N,8)
STPF 92 CCC=(SIG(N,11-SIG(N,2))*(SIG(N,5)-SIG(N,6))+(SIG(N,2(-SIG(N,
STPF 93 .4))*(SIG(N,6(-SIG(N,8)+(SIG(N,4)-SIG(N,1))*(SIG(N,8)-SIG(N,
STPF 94 .5(
STPF 95 CCC=CCC/5.0+ SIG(N,3)*SIG(N,7(
STPF 96 AR=AJ2-(FF(4(*AJ1)**2
STPF 97 BR=BJ2-(FF(4(*BJ1)**2
STPF 98 CC=(CC-(FF(4(*FE(4(*AJ1*PJ1(
STPF 99 DD=FF(4(*FE(3(*AJ1
STPF 100 FF=FE(4(*FE(3(*BJ1

```

```

STPF101      GG=FF(3)*EE(3)
STPF102      AAA=AB+BB-2.*CC
STPF103      BBB=AB-CC+DD-FF
STPF104      CCC=2.*DD-GG+AB
STPF105      GGG=BBB*BBB-AAA*CCC
STPF106      IF(GGC.LT.0.) WRITE(6,2008) N
STPF107      IF(GGG.LT.0.) GGG=DABS(GGG)
STPF108      GGG=DSQRT(GGG)
STPF109      IF(AAA.NE.0.) GO TO 220
STPF110      RATIO(N)=.5*CCC/BBB
STPF111      (D TO 300
STPF112      220 AB=BBB/AAA
STPF113      BB=DABS(GGG/AAA)
STPF114      RATIO(N)=AB-BB
STPF115      IF(RATIO(N).LT.0.) RATIO(N)=AB+BB
STPF116      IF(RATIO(N).GE.1.) RATIO(N)=.99999
STPF117      IF(RATIO(N).LT.0.) RATIO(N)=0.
STPF118      IF (KK.N(,1) GO TO 240
STPF119      WRITE (6,3000)
STPF120      240 WRITE (6,3005) N,RATIO(N)
STPF121      300 CONTINUE
STPF122      IF(KK.EQ.0) SR=1.0
STPF123      IF(KK.FQ.0) GO TO 410
STPF124      DD 350 N=1,NELMAX
STPF125      IF(MTAG(N).GT.0) GO TO 350
STPF126      MTYPE=IX(N,5)
STPF127      IF (RD(MTYPE).FQ.0.) GO TO 350
STPF128      DD=RATIO(N)
STPF129      IF(DD.GE.SR) GO TO 350
STPF130      SR=DD
STPF131      NOPT=2
STPF132      NMY=N
STPF133      KKK=1
STPF134      350 CONTINUE
STPF135      WRITE (6,3010) SR,SR2
STPF136      IF(NNN.FQ.1) GO TO 352
STPF137      DD= SR*(1.0-SR2)
STPF138      IF(DD.LT.0.03) NOPT=3
STPF139      IF(DD.LT.0.03) DD=0.03
STPF140      SR= DD/(1.0-SR2)
STPF141      DD= SR*(1.0-SR2)
STPF142      IF(DD.GT.0.10) NOPT=4
STPF143      IF(DD.GT.0.10) DD=0.10
STPF144      SR=DD/(1.-SR2)
STPF145      IF(SR.GT.1.0) SR=1.00
STPF146      352 CONTINUE
STPF147      SRATIO=SR
STPF148      IF (KJK.GT.0) SR=0.
STPF149      410 CONTINUE
STPF150      DD 420 N=1,NPMAX

```

```

STRF151      II=2*N-1
STRF152      R(11)=R(11)*SR
STRF153      420 R(11+1)=R(11+1)*SR
STRF154      WRITE (6,4010) NMY,SR,SR2
STRF155      NNN=0
STRF156      FRMAX=0.
STRF157      DO 600 N=1,NFLMAX
STRF158      MTYPE=IX(N,5)
STRF159      IF (R0(MTYPE).EQ.0.) GO TO 515
STRF160      IF (IX(N,3).NE.IX(N,2)) GO TO 430
STRF161 C
STRF162 C      CALCULATE STRESSES IN ONE DIMENSIONAL ELEMENTS
STRF163 C
STRF164      I=IX(N,1)
STRF165      J=IX(N,2)
STRF166      XC=(R(1)+R(J))/2.0
STRF167      YC=(Z(1)+Z(J))/2.0
STRF168      DX=R(J)-R(1)
STRF169      DY=Z(J)-Z(1)
STRF170      XL=DSQRT(DX**2+DY**2)
STRF171      DU=B(2*J-1)-B(2*I-1)
STRF172      DV=B(2*J)-B(2*I)
STRF173      DL=DV*DY/XL+DU*DX/XL
STRF174      DO 422 I=1,7
STRF175      422 SIG(I)=0.
STRF176      DX=(F(1,MTYPE)*DL/XL
STRF177      SIG(1)=DX*SR
STRF178      IF (F(3,MTYPE).EQ.1.) GO TO 425
STRF179      SIG(1,5)=-DX*(1-SR)
STRF180      SIG(1,N,1)=SIG(1)+SIG(1,N,1)
STRF181      GO TO 515
STRF182      425 SIG(5)=-DX*(1-SR)
STRF183      SIG(1)=SIG(1,N,1)+DX*SR
STRF184      DY=F(4,MTYPE)/(SIG(1,N,1)+DX)-1.
STRF185      IF (DAPS(DY).LE..05) GO TO 428
STRF186      II=1
STRF187      IF (NNN.GT.1) II=5
STRF188      DZ=SIG(1,N,1)+DX
STRF189      DX=SIG(1,N,1)+DX
STRF190      DY=(F(4,MTYPE)-DZ)*SIG(1,N,1)/DX
STRF191      SIG(1,N,1)=SIG(1)+DY
STRF192      SIG(1,N,5)=SIG(5)+DY
STRF193      KLK=1
STRF194      SIG(5)=SIG(1,N,5)
STRF195      SIG(4)=SIG(1,N,1)
STRF196      GO TO 515
STRF197      428 SIG(1,N,1)=SIG(1)
STRF198      SIG(1,N,5)=SIG(5)
STRF199      GO TO 515
STRF200      430 I=IX(N,1)

```



```

STRF201      J=IX(N,2)
STRF202      K=IX(N,3)
STRF203      L=IX(N,4)
STRF204      MTYPE=IX(N,5)
STRF205      IF(K.EQ.L) GO TO 440
STRF206      XC=(R(I)+R(J)+R(K)+R(L))/4.
STRF207      YC=(Z(I)+Z(J)+Z(K)+Z(L))/4.
STRF208      GO TO 445
440 STRF209      XC=(R(I)+R(J)+R(K))/3.
STRF210      YC=(Z(I)+Z(J)+Z(K))/3.
445 STRF211      CONTINUE
STRF212      XN=0.5
STRF213      IF(NNN.EQ.1) XN=1.0
STRF214      IF(MTAG(N).LE.0) XN=1.0
STRF215      DO 450 I=1,4
STRF216      II=I+4
STRF217      TT(I) = SIGI(N,I)
STRF218      TF(I)=SIGI(N,II)
STRF219      SIG(I)=SIGI(N,II)-SIGI(N,I)
STRF220      SIGI(N,II)=SIG(I)*(1.-SR)
STRF221      SIGI(N,II)=-SIGI(N,II)
STRF222      SIG(I)= XN*SIG(I)*SR+SIGI(N,I)
STRF223      450 SIGI(N,I)=SIG(I)
STRF224      IF((NMY.EQ.0).OR.(KJK.GT.0)) GO TO 485
STRF225      IF (MTAG(N).LE.0) GO TO 481
STRF226      ISTOP=0
STRF227      DO 470 JJ=1,5
STRF228      CALL STRSTR (R,Z,UR,UZ,CODE,RO,E,RO,PR,SIGI,TH,FR,IX,MTAG,
STRF229      * IRC,JBC,NUMR,NUMR1,NNP)
STRF230      DO 455 I=1,4
STRF231      II=I+4
STRF232      SIGI(N,II)=TT(I)
STRF233      DO 455 J=1,3
STRF234      455 SIGI(N,II)=SIGI(N,II)+C(I,J)*EP(N,J)*SR
STRF235      DO=(SIGI(N,5)-SIGI(N,6))*2+(SIGI(N,6)- SIGI(N,8))*2+(SIGI(N,5)-
STRF236      .SIGI(N,8))*2
STRF237      AJ2=DO/6.0+ SIGI(N,7)*2
STRF238      AJ2=DSQRT(AJ2)
STRF239      AJ1=SIGI(N,5)+SIGI(N,6)+SIGI(N,8)
STRF240      FAIL=AJ2+FE(4)*AJ1
STRF241      IF(JJ.EQ.1) GO TO 460
STRF242      D=DARS(FAIL-FAIL1)
STRF243      TOL=0.005*FAIL
STRF244      IF(DO.LE.TOL) ISTOP=1
STRF245      460 FAIL1 =FAIL
STRF246      DO 465 I=1,4
STRF247      II=I+4
STRF248      SIG(I)=SIGI(N,II)-TT(I)
STRF249      465 SIGI(N,I)=TT(I)+SIG(I)*XN
STRF250      IF(N.NE.1) GO TO 451

```

```

STRF251      WRITE(6,1000) (SIGI(N,I),I=1,8)
STRF252 1000 FORMAT(10H SIGI(N,I) / (PF12.4))
STRF253      451 CONTINUE
STRF254      IF(ISTOP.GT.0) GO TO 475
STRF255      470 CONTINUE
STRF256      475 DO 480 I=1,4
STRF257          II=I+4
STRF258          SIGI(N,I)=SIGI(N,II)
STRF259          SIGI(N,II)=-(TF(I)-SIGI(N,I))
STRF260          SIG(I)=SIGI(N,I)
STRF261      480 CONTINUE
STRF262      481 TR=SRATIO
STRF263          TOL=0.05*TR
STRF264          IF(TR.GE.0.95) GO TO 485
STRF265          IF(TOL.LT.0.005) TOL=0.005
STRF266          IF(MTAG(N).GT.0) GO TO 485
STRF267          IF(RATIO(N).LE.TR) GO TO 482
STRF268          DD=PATIO(N)-TR
STRF269          IF(DD.GT.TOL) GO TO 485
STRF270      482 NDD=NDD+1
STRF271          NFW(NDD)=N
STRF272          MTAG(N)=1
STRF273      485 CONTINUE
STRF274          DO 490 I=1,4
STRF275      490 FF(I)=F(I,MTYPE)
STRF276          CC= 2.*DSIN(FF(4)/57.296)
STRF277          RR= 1.732*(3.-DSIN(FF(4)/57.296))
STRF278          PP=6.*DCOS(FF(4)/57.296)
STRF279          FF(4)=CC/8R
STRF280          EE(3)= EE(3)*PP/RR
STRF281          SIG(7)=SIGI(N,4)
STRF282          CC=(SIG(1)+SIG(2))/2.
STRF283          RR=(SIG(1)-SIG(2))/2.
STRF284          CR=DSQRT(PR**2+SIG(3)**2)
STRF285          SIG(4)=CC+CR
STRF286          SIG(5)=CC-CR
STRF287          SIG(6)=0.0
STRF288          IF ((BR.EQ.0.0).AND.(SIG(3).EQ.0.0)) GO TO 510
STRF289          SIG(6)=2R.64R*DATAN2(SIG(3),RR)
STRF290      510 CONTINUE
STRF291          DD=(SIG(1)-SIG(2))**2+(SIG(2)-SIG(7))**2+(SIG(7)-SIG(1))**2
STRF292          AJ2=DD/6. +SIG(3)**2
STRF293          AJ2=DSQRT(AJ2)
STRF294          AJ1=SIG(1)+SIG(2)+SIG(7)
STRF295          FAIL=FF(3)-FF(4)*AJ1
STRF296          FR(N)=AJ2/FAIL
STRF297          IF (MTAG(N).GT.0) GO TO 515
STRF298          IF (KJK.GT.0) GO TO 515
STRF299          IF (FR(N).GE.0.99) MTAG(N)=1
STRF300          IF (MTAG(N).EQ.0) GO TO 515

```

```

STRE301      NOO=NOO+1
STRE302      NEW(NOO)=N
STRE303      515 IF(MPRINT.GT.0) GO TO 550
STRE304      WRITE(6,2000) NNN
STRE305      MPRINT=50
STRE306      550 MPRINT=MPRINT-1
STRE307      IF((IX(N,2)-IX(N,3)).EQ.0) FR(N)=0.0
STRE308      IF(TH(N).NE.1.) SIG(7)=0.0
STRE309      IF (RO(MTYPE).NF.0.) GO TO 555
STRE310      WRITE (6,2010) N
STRE311      GO TO 600
STRE312 C
STRE313 C      PRINT OUT STRESSES
STRE314 C
STRE315      555 IF(MTAG(N).LT.1) GO TO 556
STRE316      WRITE(6,2011) N,XC,YC,(SIG(I),I=1,7),MTAG(N),FR(N),MTYPE
STRE317      GO TO 557
STRE318      556 WRITE(6,2001) N,XC,YC,(SIG(I),I=1,7),MTAG(N),FR(N),MTYPE
STRE319      557 IF(IX(N,2).EQ.IX(N,3)) GO TO 600
STRE320      IF((FR(N).GE.1.) CR.(FR(N).LT.FRMAX)) GO TO 570
STRE321      FRMAX=FR(N)
STRE322      NMAX=N
STRE323      570 CONTINUE
STRE324      DO=SIGI(N,5)-SIGI(N,6)
STRE325      FF=SIGI(N,6)-SIGI(N,8)
STRE326      GG=SIGI(N,5)-SIGI(N,8)
STRE327      BJ2= (DO**2+GG**2+FF**2)/6.0 +SIGI(N,7)**2
STRE328      BJ2=DSQRT(BJ2)
STRE329      TOLL=TOLL+BJ2
STRE330      600 CONTINUE
STRE331      SR2= (SR1*SR) + SR2
STRE332      SR1= (1.0-SR) * SR1
STRE333      WRITE(6,2002) TOLL,SR,NMY,KK,SR2
STRE334      WRITE(6,5000) FRMAX,NMAX
STRE335      IF(NOO.EQ.0) GO TO 620
STRE336      WRITE(6,2020) (NEW(I),I=1,NOO)
STRE337      620 KKK=1
STRE338      IF(TOLL.LE.TOLLA) KKK=0
STRE339      NKT=NFLMAX/2
STRE340      JCK=0
STRE341      DO 705 N=1,NFLMAX
STRE342      IF(MTAG(N).GT.0) JCK=JCK+1
STRE343      705 CONTINUE
STRE344      IF(JCK.LT.NKT) GO TO 710
STRE345      WRITE(6,2007)
STRE346      CALL EXIT
STRE347      710 CONTINUE
STRE348      IF(NNN.EQ.1) GO TO 800
STRE349      GO TO (750,760,770,780),NOPT
STRE350      GO TO 800

```

```

STRE351 750 WRITE(6,2003) JJJ
STRE352 GO TO 800
STRE353 760 WRITE(6,2004)
STRE354 GO TO 800
STRE355 770 WRITE(6,2005)
STRE356 GO TO 800
STRE357 780 WRITE(6,2006)
STRE358 800 RETURN
STRE359 2000 FORMAT(1H1/
STRE360 .36H STRESSES AFTER APPROXIMATION NUMBER 14////
STRE361 .7H FL.NO. 7X 1HX 7X 1HY 4X 8HX-STRESS 4X 8HY-STRESS 3X 9HXY-STRESS
STRE362 . 2X 10HMAX-STRESS 2X 10HMIN-STRESS 7H ANGLE 4X 8HZ-STRESS 3X 7HPL
STRE363 .ASTIC 3X 4HFAIL 3X 5HMYRF )
STRE364 2001 FORMAT (17,2FR.2,1P5E12.4,0P1F7.2,1PF12.4, 16,0P1F11.3 ,16 1
STRE365 2002 FORMAT(39H0THE UNBALANCED LOAD AT THIS STAGE IS F14.5//
STRE366 .47H THE RATIO FOR CORRECTION OF STORED STRESSES IS F10.4//
STRE367 .31H THE NEXT ELEMENT YIELDING IS 14/
STRE368 .91H AND THE TOTAL NUMBER OF ELEMENTS THAT CAN YIELD WITH THE LINEA
STRE369 .R ADDITION OF TOTAL LOAD IS 14/
STRE370 .50H LOAD UP TO THIS STAGE AS A FRACTION OF TOTAL IS F20.5 1
STRE371 2003 FORMAT(110H0STRESS RATIO GOVERNED BY STRESS STATE BEING MORE THAN
STRE372 .5 PERCENT OUTSIDE YIELD SURFACE FOR ELEMENT NUMBER = 15)
STRE373 2004 FORMAT( 52H0 STRESS RATIO GOVERNED BY NEXT ELEMENT YIELDING )
STRE374 2005 FORMAT(100H0STRESS RATIO GOVERNED BY THE MINIMUM VALUE OF SR FOR
STRE375 . ANY STEP )
STRE376 2006 FORMAT(110H0 STRESS RATIO GOVERNED BY THE MAXIMUM VALUE OF STRESS
STRE377 . RATIO FOR ANY STEP )
STRE378 2007 FORMAT(64H0 JOB TERMINATED AS HALF OF TOTAL ELEMENTS YIELD AT THIS
STRE379 . STEP )
STRE380 2008 FORMAT( 110H ARGUMENT NEGATIVE IN THE EQUATION FOR CALCULATING
STRE381 . THE VALUE OF STRESS RATIO FOR ELEMENT NUMBER = 151
STRE382 2009 FORMAT(45H0 STRESS RATIO SR FOR THIS CYCLE= F10.5)
STRE383 2010 FORMAT(17,50H THIS ELEMENT HAS BEEN REMOVED FROM THE ANALYSIS )
STRE384 2011 FORMAT (17,2FR.2,1R5E12.4,0P1F7.2,1PF12.4, 16,0P1F11.3 ,16,3X,1H*)
STRE385 2020 FORMAT(4PH0 THE FOLLOWING NEW ELEMENTS YIELD IN THIS STEP //
STRE386 .20(5)
STRE387 3000 FORMAT (35H0 THE FOLLOWING ELEMENTS CAN YIELD //
STRE388 .34H0 ELEMENT NO. RATIO(N) )
STRE389 3005 FORMAT ( (10,F20.5)
STRE390 3010 FORMAT( 17H0 SR S92 /
STRE391 . 2F10.5)
STRE392 4010 FORMAT(6H0 NMY= 15,3HSR= F20.5, 4HSR2= E20.5 )
STRE393 5000 FORMAT(17H0MAXIMUM FAIL (S F6.3,17H FOR ELEMENT NO. (5)
STRE394 END

```

CHAPTER III: GRIFTH - A Computer Program for Two-Dimensional
Analysis of Progressive Failure of Rock Following
Griffith and Modified Griffith Theory

3.1. Purpose and Capability

This computer program is applicable to plane stress or plane strain analysis of stresses, deformations and progressive fracture in elastic brittle rock following Griffith and modified Griffith theory. Arbitrary initial stresses, arbitrary sequence of construction or excavation, arbitrary history of load application can be simulated. One dimensional elements are included. The program is applicable to study of fracture initiation and propagation in arbitrary elastic brittle structure systems composed of several different materials. Non-monotonic loading is considered.

Theoretical development incorporated in the program is documented in Part I- Technical Report of this report.

3.2. Program Organization

The computer program is in Fortran language. Files 1 and 2 are used to store system equations and element properties respectively. Tapes 5 and 6 are the input/output files. The program capacity can be altered by changing the dimensions of arrays AA and IA. These correspond to the total locations required for real and integer arrays respectively. NTOT, MTOT at lines MAIN 29 , MAIN 30 are set equal to the dimensions of AA and IA.

The program consists of the following units:

a. MAIN

In this unit, the control information including maximum number of elements,

nodal points, different materials, boundary pressure cards, the number of steps of excavation or construction, the maximum number of elements removed or added to the system at any stage is read in. This information is used to organize the dimensions of various arrays. This done, further processing of data is done in subroutine INPT.

b. Subroutine INPT

This subroutine is called by the unit MAIN. The first step is to read in material property data for all different materials in the system. Nodal point coordinates, loads and code descriptors for boundary conditions are read in or generated. Element geometry, initial stresses, initial crack openings, if any, are read in or generated. Maximum band width for the system is calculated and dimensions of blocks for generation and storage of system stiffness defined. After defining these controls, the incremental structure is analyzed in steps. For each step the number of nodal points, the number of elements, the number of elements and nodal points removed or added, if any, the number of boundary pressure cards and the material type of the elements added or changes in material properties are read in. After the information is assembled the solution process is transferred to subroutine SOLVE.

c. Subroutine SOLVE

This subroutine called by INPT is concerned with obtaining stresses, deformations and sequence of progressive fracture of elements in a given step of loading/construction/excavation. To trace progressive fracture, the solution process traces a sequence of elements reaching fracture along with the effects of stress redistribution

associated with fracture leading to secondary fractures. The process consists of applying the total load and then scaling it according to the minimum ratio of load increment needed to ensure one element reaching fracture. Once an element fractures, the associated stress redistribution will result in secondary fractures at the same total load. This is referred to as system stability iteration in the program.

The SOLVE subroutine calls ONED and QUAD to obtain element stiffness for one or two-dimensional elements respectively. This information is stored on File 2 and is updated in case of fracture or failure. Solution to the stiffness equations is obtained in subroutine BANSOL. Subroutine STRESS defines the stresses corresponding to a load application. This is referred to as the initial state for any load increment. As several elements may crack during a load increment, it is necessary to scale it to pin-point the sequential fracture phenomenon. This is accomplished in subroutine SCALE.

d. Subroutine SCALE

This subroutine calls subroutine GRIFTH to check each element for fracture or closure of cracks. The stress ratio for each element is calculated, if necessary, using interpolation (subroutine INTER). Iterations to define a value of orientation of fracture are referred to as beta-stability iterations. After choosing the minimum stress ratio applicable and the element that next fractures, system stability iteration is accomplished to define all secondary fractures associated with the primary fracture. This process is illustrated in the flow chart as Figure III-1.

3.3. Input Data

a. Job Title (18A4). This card will give the descriptive identification for the job.

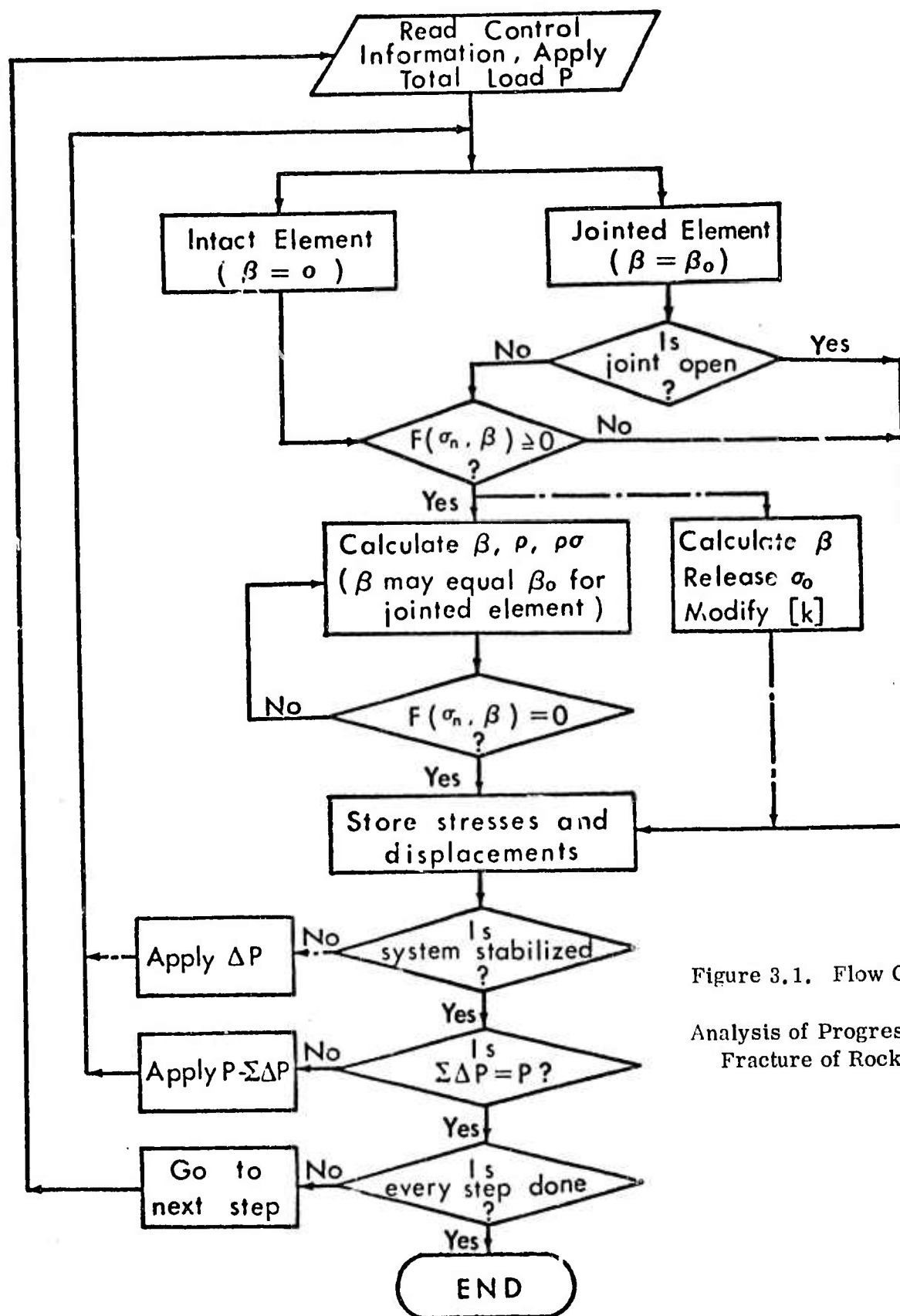


Figure 3.1. Flow Chart
Analysis of Progressive
Fracture of Rock

b. Control Information (4I5, 3F10.2, 3I5)

<u>Information</u>	<u>Columns</u>
Maximum number of nodal points	1-5
Maximum number of elements	6-10
Number of different materials	11-15
Maximum number of pressure cards	16-20
Body force in X-direction	21-30
Body force in Y-direction	31-40
Reference (stress-free) temperature	41-50
Number of initially open cracks	51-55
Code to designate plane stress or plane strain NPLANE = 1 for plane stress = 2 for plane strain	56-60
Total number of excavation & construction steps	61-65

c. Material Property Cards. One set of cards must be provided for each material.

In each set:

i. First card (2I5, F10.3, 2I5) will give the following information:

<u>Information</u>	<u>Columns</u>
Material identification number	1-5
Number of temperature cards (8 maximum)	6-10
Mass density of the material	11-20
Material code to designate materials which will follow the fracture criteria	21-25

code = 1 for materials which will follow the fracture criteria
 = 0 for materials which will not fracture

Material code to identify the initial anisotropy due to initial cracks 26-30
 code = 1 for initial anisotropy
 = 0 for isotropy

ii. If columns 21-25 in card i is not zero, the following information must be provided (3F10.3)

<u>Information</u>	<u>Columns</u>
Tensile strength	1-10
Internal frictional coefficient	11-20
Tolerance for crack closure	21-30

iii. Subsequent cards, one for each temperature, the number being defined in columns 6-10 of the first card, will carry the following information (F10.0, E10.0, 2F10.0):

<u>Information</u>	<u>Columns</u>
Temperature	1-10
Elastic modulus	11-20
Poisson's ratio	21-30
Coefficient of thermal expansion	31-40

iv. If columns 26-30 in card i is not zero, the following information must be provided (4F10.4):

<u>Information</u>	<u>Columns</u>
Modulus ratio	1-10
Angle of fault	11-20
Frictional coefficient	21-30
Shear strength	31-40

- d. Nodal Point Cards (I5, F5.0, 5F10.0). One card for each nodal point with the following information:

<u>Information</u>	<u>Columns</u>
Nodal point number	1-5
Type of nodal point	6-10
X-ordinate	11-20
Y-ordinate	21-30
XR	31-40
XZ	41-50
Temperature	51-60

If the number in columns 6-10 is:

0 = XR is the specified X-load and XZ is the specified Y-load

1 = XR is the specified X-displacement and XZ is the specified Y-load

2 = XR is the specified X-load and XZ is the specified Y-displacement

3 = XR is the specified X-displacement and XZ is the specified Y-displacement.

All loads are considered to be total forces acting on an element of unit thickness.

Nodal point cards must be in numerical sequence. If cards are omitted, the omitted nodal points are generated at equal intervals along a straight line between the defined nodal points. The necessary temperatures are determined by linear interpolation.

The type of the nodal point, as well as XR, XZ, are set equal to zero.

- e. Element Material Cards (16I5). These cards shall carry the material type of all the elements. Each card shall have material types for 16 elements in sequence. The material type for each element must be read in as no interpolation has been provided for.

- f. Elements Cards (5I5, 5X, 3F10.0). One card for each element will provide the following data:

<u>Information</u>	<u>Columns</u>
Number of element	1-5
Nodal point I	6-10
Nodal point J	11-15
Nodal point K	16-20
Nodal point L	21-25
Initial stresses:	
Component in X-direction	31-40
Component in Y-direction	41-50
Shearing stress on X-Y planes	51-60

Nodal points I, J, K, L are corners of each individual element in a counter-clockwise order for a right handed system of coordinates. For triangular elements set nodal point L same as nodal point K. The element cards must be in the numerical sequence. Any cards that are omitted will be automatically generated in the program by incrementing each of the I, J, K and L nodal points by one. The material type will be taken same as for the last element defined.

- g. Initial Cracks Cards. If columns 41-45 in card b is not zero, the following information must be provided for each initial crack (2I5, F10.0).

<u>Information</u>	<u>Columns</u>
Element number	1-5

Tag number 6-10

Crack Angle 11-20

The tag number is used to designate the crack mode

Tag - 1 for single crack
2 for double crack

The crack angle is defined as the angle between X-axis and the normal to the crack plane, positive counter-clockwise.

h. Incremental Step Information. One set of cards must be provided for each step of construction or dismantling. Construction and dismantling may not be mixed in one step.

- i. First card (18A4). This gives the descriptive title for the step for which information follows.
- ii. Second card (7I5). Following information is given for the step which is described in Title.

<u>Information</u>	<u>Columns</u>
Number of nodal points in this step	1-5
Number of elements in this step	6-10
Number of elements removed or added with reference to previous step	11-15
Number of pressure boundary cards (total pressure for this step)	16-20
New material type for the elements added or eliminated	21-25
Code to designate addition to structure or dismantling	26-30

Code = 0 for dismantling
= 1 for construction

Number of nodal points included in analysis but not part of structure at this step

31-35

- iii. Addition or Dismantling of Structure. One or more cards shall indicate the numbers of elements removed or added in the step under consideration (16I5).

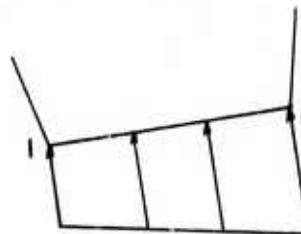
The total number of elements included here must be the same as indicated in columns 11-15 of card h-ii.

- iv. Nodal Points Included in Analysis but not Taking Any Load (16I5). One or more cards shall indicate the numbers of the nodal points included in the analysis at this step but not forming part of the load carrying system.

The total number of nodal points listed here must be the same as indicated in columns 31-35 of card h-ii.

- v. Pressure Boundary Cards (2I5, 2F10.0). If there are any boundary pressures for this step, then one card for each boundary element which is subjected to normal pressure will carry the following information:

<u>Information</u>	<u>Columns</u>
Nodal Point I	1-5
Nodal Point J	6-10
Total Normal Pressure at I	11-20
Total Normal Pressure at J	21-30



As shown in the sketch, the boundary element must be on the left as one progresses from I to J. Surface tensile force is input as a negative pressure.

- j. Last Card (A6). The last card at the end of data deck is "stop" card. It carries the characters STOP in columns 1 through 4.

3.4. Output Information

The following information is developed and printed by the program:

- a. Print out of problem data. This includes information in material properties, mesh layout, geometry and boundary conditions, loads and constraints, initial cracks, etc.
- b. Initial stresses before any excavation or construction at the center of each element.
- c. The incremental as well as cumulative nodal point displacements and the stresses after application of a load increment.
- d. The stresses upon application of total load increment.
- e. The stresses, crack orientation, type of crack, at the first element cracking under a load increment.
- f. The stresses, crack orientation, type of crack for each beta-iteration to define the correct crack orientation.
- g. The stresses, crack orientation, type of crack for each system stability iteration to define secondary fractures following the first fracture in a load increment .
- h. The stress ratio as a proportion of total load at each stability iteration.

3.5. Fortran Listing

```

MAIN 1 C *****
MAIN 2 C * PROGRAM IDENTIFICATION: PROGRAM PFA *
MAIN 3 C * PROGRAMMER: S.W. HUANG, I.S. RAI, THE OHIO STATE UNIVERSITY *
MAIN 4 C * PURPOSE: PROGRESSIVE FRACTURE ANALYSIS *
MAIN 5 C * FAILURE CRITERIA: FRACTURE ACCORDING TO GRIFFITH AND MODIFIED *
MAIN 6 C * GRIFFITH THEORIES *
MAIN 7 C * THE FORMULATION IS DOCUMENTED IN THE FINAL REPORT DATED MARCH *
MAIN 8 C * 31, 1973, ON CONTRACT HC210017 BETWEEN THE OHIO STATE UNIVERSITY *
MAIN 9 C * AND THE UNITED STATES BUREAU OF MINES SUPPORTED BY THE ADVANCED *
MAIN 10 C * RESEARCH PROJECTS AGENCY. INSTRUCTIONS FOR USE OF THE PROGRAM *
MAIN 11 C * ARE CONTAINED IN PART II OF THE REPORT. *
MAIN 12 C *****
MAIN 13 C
MAIN 14 COMMON AAI160001,IA(4500)
MAIN 15 COMMON/ONE/ NUMNP,NUMEL,NUMMAT,NUMPC,NPC,MBAND,NUMBLK,NL,MTYPE,N,
MAIN 16 . VUL,ACELR,ACELZ,Q,HEDI18),STOP,SR,SR1,TOTAL,TOL,TOL1,XC,YC,
MAIN 17 . TEMP,SIGN,SIGD1,SIGD2,
MAIN 18 . LLL,111,JJJ,JCK,KCHECK,JCHECK,NPLANE,NUMER,NCRACK,JA,NFQ
MAIN 19 . ,NCRACK,NSTEP,N15,M7,NTOT
MAIN 20 COMMON/TWO/ C13(3),S(10,101,SIG(6),PI8),ST(3,10),PR(51,2215),
MAIN 21 . LM(4),FF131,FP5(31
MAIN 22 COMMON/THREE/ E(8,4,8),EDI4,8),TENS(8),XNU(8),RU(81-EPST(8),
MAIN 23 . MTC(8),NIC16),MID(8)
MAIN 24 DEFINE FILE 1(100,1500,U,NBK),2(510,230,U,10)
MAIN 25 CALL ERRSET(208,256,-1,11)
MAIN 26 C
MAIN 27 C READ AND WRITE CONTROL INFORMATION FOR THE PROBLEM
MAIN 28 C
MAIN 29 NTOT=16000
MAIN 30 MTOT=4500
MAIN 31 TOL1=0.001
MAIN 32 READ 15,1000) HED,NUMNP,NUMEL,NUMMAT,NUMPC,ACELR,ACELZ,Q,NCRACK,
MAIN 33 . NPLANE,NSTEP
MAIN 34 IF(NPLANE.EQ.1) WRITE(6,20001
MAIN 35 IF(NPLANE.EQ.2) WRITE(6,20005)
MAIN 36 NEQ=2*NUMNP
MAIN 37 WRITE 16,20101 HED,NUMNP,NUMEL,NUMMAT,NUMPC,ACELR,ACELZ,Q,NSTEP
MAIN 38 NPC=NUMPC
MAIN 39 IF(NPC.EQ.0) NPC=1
MAIN 40 N1=1
MAIN 41 N2=N1+NUMNP
MAIN 42 N3=N2+NUMNP
MAIN 43 N4=N3+NUMNP
MAIN 44 N5=N4+NEQ
MAIN 45 N6=N5+NEQ
MAIN 46 N7=N6+NEQ
MAIN 47 N8=N7+NEQ
MAIN 48 N9=N8+NUMNP
MAIN 49 N10=N9 +6*NUMEL
MAIN 50 N11=N10+6*NUMEL

```

```

MAIN 51      N12=N11+ NUMEL
MAIN 52      N13=N12+ NUMEL
MAIN 53      N14=N13+ NUMEL
MAIN 54      N15=N14+2*NP1
MAIN 55      M1=1
MAIN 56      M2=M1+5*NUMEL
MAIN 57      M3=M2+NUMEL
MAIN 58      M4=M3+NUMEL
MAIN 59      M5=M4+NUMEL
MAIN 60      M6=M5+NPC
MAIN 61      M7=M6+NPC
MAIN 62      JJ=M7-MT01
MAIN 63      IF(JJ.LT.0) GO TO 100
MAIN 64      WRITE (6,3000) JJ
MAIN 65      CALL EXIT
MAIN 66      100 CONTINUE
MAIN 67      CALL INPT (AA(N1),AA(N2),AA(N3),AA(N4),AA(N5),AA(N6),AA(N7),
MAIN 68      .      AA(N8),AA(N9),AA(N10),AA(N11),AA(N12),AA(N13),AA(N14),IA(M1),
MAIN 69      .      IA(M2),IA(M3),IA(M4),IA(M5),IA(M6))
MAIN 70      STOP
MAIN 71      1000 FORMAT (18A4/4I5,3F10.2,3I5)
MAIN 72      2000 FORMAT(1H1,' PLANE STRESS ANALYSIS')
MAIN 73      2005 FORMAT(1H1,' PLANE STRAIN ANALYSIS')
MAIN 74      2010 FORMAT (1H0 18A4/
MAIN 75      1 30H0 NUMBER OF NODAL POINTS----- 13 /
MAIN 76      2 30H0 NUMBER OF ELEMENTS----- 13 /
MAIN 77      3 30H0 NUMBER OF DIFF. MATERIALS--- 13 /
MAIN 78      4 30H0 NUMBER OF PRESSURE CARDS--- 13/
MAIN 79      5 30H0 X-ACCELERATION----- E12.4/
MAIN 80      6 30H0 Y-ACCELERATION----- E12.4/
MAIN 81      7 30H0 REFERENCE TEMPERATURE----- E12.4/
MAIN 82      8 30H0 NUMBER OF STEPS----- 13 )
MAIN 83      3000 FORMAT (70H PROGRAM EXECUTION TERMINATED. REQUIRED CORE EXCEEDS MT
MAIN 84      .GT BY 110)
MAIN 85      END

```

```

INPT 1      SUBROUTINE (NPT, (R,Z,CODE,UU,CU,FI,BJ,I,SIGI,EPSI,SIGNM,BETA,
INPT 2      . RATIO,PR,IX,MIAG,NTAG,JNT),(BC,JBC)
INPT 3 C
INPT 4      COMMON AA(10000),(A(4500)
INPT 5      COMMON/ONE/ NUMNP,NUMEL,NUMMAT,NUMPC,NPC,MBAND,NUMELK,NL,MTYPE,N,
INPT 6      . VOL,ACFLK,ACIL2,Q,HEDT18),STOP,SR,SRI,TOTAL,TOL,ICL(XC,YC,
INPT 7      . TEMP,SIGN,SIGDI,SIGDJ,
INPT 8      . LLL,III,JJJ,JCK,KCHECK,JCHECK,NPLANE,NUMER,NODE,JA,NFO
INPT 9      . ,NLAND,NCRACK,NSTEP,NIS,M7,NTUT
INPT 10     COMMON/TWO/ C(3,3),S(10,10),SIG(6),P(6),ST(3,10),RR(5),ZZ(5),
INPT 11     . LM(4),FE(3),EPS(3)
INPT 12     COMMON/THREE/ L(8,4,8),EO(4,8),TENS(8),XNU(8),RO(8),EPST(8),
INPT 13     . MTC(8),NTC(8),MID(8)
INPT 14     DIMENSION TITLE(16),NNP(200),MAD(200),NUMR(200)
INPT 15     DIMENSION R(NUMNP),Z(NUMNP),CODE(NUMNP),UU(NUMNP,2),CU(NFO),
INPT 16     .BI(NFO) (NFO),I(NUMNP),SIGI(NUMEL,6),EPSI(NUMEL,6),SIGNM(NUMEL),
INPT 17     .BETA(NL),RATIO(NUMEL),PR(NPC,2),IX(NUMEL,5),MIAG(NUMEL),
INPT 18     .NTAG(NUM),JNT(NUMEL),(BC(NPC),JBC(NPC)
INPT 19     DO 50 M=1,NUMMAT
INPT 20     READ (5,1010) MTYPE,NTC(MTYPE),RO(MTYPE),MTC(MTYPE),MID(MTYPE)
INPT 21     WRITE(6,2010) MTYPE,NTC(MTYPE),RO(MTYPE),MTC(MTYPE),MID(MTYPE)
INPT 22     IF (MTC(MTYPE).EQ.0) GO TO 45
INPT 23     READ(5,1015) TENS(MTYPE),XNU(MTYPE),EPST(MTYPE)
INPT 24     WRITE(6,2015) TENS(MTYPE),XNU(MTYPE),EPST(MTYPE)
INPT 25 45 CONTINUE
INPT 26     NUMTC=NTC(MTYPE)
INPT 27     READ (5,1020) ((E(I,J,MTYPE),J=1,4),I=1,NUMTC)
INPT 28     WRITE (6,2020) ((E(I,J,MTYPE),J=1,4),I=1,NUMTC)
INPT 29     IF (MID(MTYPE).EQ.0) GO TO 50
INPT 30     READ (5,1022) (EO(K,MTYPE),K=1,4)
INPT 31     WRITE(6,2022) (EO(K,MTYPE),K=1,4)
INPT 32     EO(2,MTYPE)=EO(2,MTYPE)/57.296
INPT 33 50 CONTINUE
INPT 34 C
INPT 35 C      READ NODAL POINT DATA,GENERATE INTERMEDIATE POINTS AND WRITE
INPT 36 C
INPT 37     WRITE (6,2025)
INPT 38     L=0
INPT 39 60 READ (5,1025) N,CODE(N),R(N),Z(N),UU(N,1),UU(N,2),I(N)
INPT 40     IF (N.EQ.1) GO TO 70
INPT 41     ZX=N-L
INPT 42     DX=(R(N)-R(L))/ZX
INPT 43     DZ=(Z(N)-Z(L))/ZX
INPT 44     DT=(I(N)-I(L))/ZX
INPT 45 70 L=L+1
INPT 46     IF (N-L) 100,90,80
INPT 47 80 CODE(L)=0.0
INPT 48     R(L)=R(L-1)+DX
INPT 49     Z(L)=Z(L-1)+DZ
INPT 50     I(L)=I(L-1)+DT

```

```

INPT 51      UU(L,1)=0.
INPT 52      UU(L,2)=0.
INPT 53      GO TO 70
INPT 54      90 IF (NUMNP-N) 100,110,60
INPT 55      100 WRITE (6,2035) N
INPT 56      CALL EXIT
INPT 57      110 WRITE(6,2030) (N, CODE(N), R(N), Z(N), UU(N,1), UU(N,2), T(N), N=1, NUMNP)
INPT 58 C
INPT 59 C      READ ELEMENT DATA , GENERATE INTERMEDIATE ELEMENTS AND WRITE
INPT 60 C
INPT 61      READ(5,1030) (IX(N,5), N=1, NUMEL)
INPT 62      WRITE (6,2040)
INPT 63      N=0
INPT 64      130 READ(5,1035) M, (IX(M,1), I=1,4)
INPT 65      140 N=N+1
INPT 66      IF (M-N) 170,170,150
INPT 67      150 IX(N,1)=IX(N-1,1)+1
INPT 68      IX(N,2)=IX(N-1,2)+1
INPT 69      IX(N,3)=IX(N-1,3)+1
INPT 70      IX(N,4)=IX(N-1,4)+1
INPT 71      170 CONTINUE
INPT 72      SIGI(N,1)=0.
INPT 73      SIGI(N,2)=0.
INPT 74      SIGI(N,3)=0.
INPT 75      MTAG(N)=0
INPT 76      JNT(N)=0
INPT 77      BETA(N)=0.
INPT 78      IF (M-N) 180,180,140
INPT 79      180 IF (NUMEL-N) 190,190,130
INPT 80      190 WRITE (6,2045) (N, (IX(N,1), I=1,5), (SIGI(N,1), I=1,3), N=1, NUMEL)
INPT 81      IF (NCRACK.EQ. 0) GO TO 194
INPT 82      WRITE(6,2050)
INPT 83      DO 192 M=1, NCRACK
INPT 84      READ (5,1040) N, MTAG(N), BETA(N)
INPT 85      WRITE(6,1040) N, MTAG(N), BETA(N)
INPT 86      192 BETA(N)=BETA(N)/57.296
INPT 87      194 CONTINUE
INPT 88 C
INPT 89 C      INITIALIZE STRAINS
INPT 90 C
INPT 91      DO 196 N=1, NUMEL
INPT 92      NTAG(N)=0
INPT 93      DO 196 I=1,3
INPT 94      196 EPSI(N,I)=0.0
INPT 95 C
INPT 96 C      INITIALIZE CUMULATIVE DISPLACEMENTS
INPT 97 C
INPT 98      DO 198 I=1, NEQ
INPT 99      198 CU(I)=0.0
INPT 100     JJ=0

```



```

INPT101      DO 199 I=1,NUMFL
INPT102      DO 199 J=1,4
INPT103      DO 199 K=1,4
INPT104      KK=IABS(IX(1,J)-IX(1,K))
INPT105      IF (KK.GT.JJ) JJ=KK
INPT106      199 CONTINUE
INPT107      NBAND=2*JJ+2
INPT108      NL=(NTOT-N15+1)/(NBAND+1)
INPT109      NLL=N1Q+3
INPT110      IF (NL.GT.NLL) NL=NLL
INPT111      NL=NL/4
INPT112      ND=2*NL
INPT113      NL=4*NL
INPT114      IF (ND.GE.NBAND) GO TO 209
INPT115      WRITE (6,4010) NL,NBAND
INPT116      CALL EXIT
INPT117      209 N16=N15+NL
INPT118      N17=N16+NL*NBAND
INPT119      JJ=N17-NTOT
INPT120      IF (JJ.LE.0) GO TO 210
INPT121      WRITE (6,3000) JJ
INPT122      CALL EXIT
INPT123      210 WRITE (6,4000) N17,M7,NL,NBAND
INPT124 C
INPT125 C      READ CONTROL INFORMATION FOR THE NEXT STEP IN INCREMENTAL ANALYSIS
INPT126 C
INPT127 C      NUMBER OF NODAL POINTS MISSING NPMIS IS INTRODUCED TO TAKE
INPT128 C      CARE OF EMPTY NODAL POINTS IN THE PREVIOUS STEP AND NEW NODAL
INPT129 C      POINTS INTRODUCED AS A CONSEQUENCE OF CONSTRUCTION OR EXCAVATION
INPT130 C      DONE DURING THIS STEP. NPMIS IS TAKEN AS THE SUM OF EMPTY NODAL
INPT131 C      POINTS DURING PREVIOUS STEP PLUS THE NODAL POINTS VACATED OR
INPT132 C      ADDED IN THIS STEP UPTO NPMAX .
INPT133 C
INPT134      NCASE=0
INPT135      200 READ(5,1045) TITLE
INPT136      WRITE(6,2055) TITLE
INPT137      READ(5,1050) NPMAX, NELMAX, NUMER, NUMPC, MTYPE, NCODE, NPMIS, MAND
INPT138      LLL=1
INPT139      III=0
INPT140      JJJ=0
INPT141      TOTAL=0.0
INPT142      SIG01=0.
INPT143      SIG0J=0.
INPT144      202 IF (NUMER.EQ.0) GO TO 204
INPT145      READ(5,1030) (NUMR(N), N= 1,NUMER)
INPT146      204 IF (NPMIS.EQ.0) GO TO 206
INPT147      READ(5,1030) (NNP(M), M= 1,NPMIS)
INPT148 C
INPT149 C      DETERMINE BANDWIDTH
INPT150 C

```

```

INPT151 206 NUMEL=NF(MAX
INPT152      NUMNP=NPMAX
INPT153      J=0
INPT154      DO 208 N=1,NUMEL
INPT155      DO 208 I=1,4
INPT156      DO 208 L=1,4
INPT157      KK=1ABS(IX(N,I)-IX(N,L))
INPT158      IF (KK-J) 208,208,207
INPT159      207 J=KK
INPT160      208 CONTINUE
INPT161      MBAND=2*J+2
INPT162      IF (NPMIS.EQ.0) GO TO 215
INPT163      DO 212 I = 1, NPMIS
INPT164      J = NNP(I)
INPT165      CU(2*J-1) = 0.
INPT166      212 CU(2*J) = 0.
INPT167 C
INPT168 C      PRINT OUT CONTROL INFORMATION FOR THE CURRENT STEP IN INCREMENTAL
INPT169 C      ANALYSIS
INPT170 C
INPT171      215 IF(NCODE.EQ.1) GO TO 240
INPT172      IF(NUMER.EQ.0) GO TO 220
INPT173      WRITE(6,2060) NCASE,NPMAX,NELMAX,NUMPC,NPMIS,(NUMR(N),N=1,NUMR )
INPT174      GO TO 230
INPT175      220 WRITE (6,2060) NCASE,NPMAX,NELMAX,NUMPC,NPMIS
INPT176      230 IF(NPMIS.EQ.0) GO TO 270
INPT177      WRITE(6,2065) (NNP(M),M = 1,NPMIS)
INPT178      GO TO 270
INPT179      240 IF(NUMER.EQ.0) GO TO 250
INPT180      WRITE(6,2070) NCASE,NPMAX,NELMAX,NUMPC,NPMIS,(NUMR(N),N=1,NUMR )
INPT181      GO TO 260
INPT182      250 WRITE(6,2070) NCASE,NPMAX,NELMAX,NUMPC,NPMIS
INPT183      260 IF(NPMIS.EQ.0) GO TO 270
INPT184      WRITE(6,2065) (NNP(M),M = 1,NPMIS)
INPT185 C
INPT186 C      IF THERE ARE ANY BOUNDARY PRESSURES FOR THIS STEP READ AND PRINT
INPT187 C      THIS DATA
INPT188 C
INPT189      270 IF (NUMPC) 290,310,290
INPT190      290 WRITE (6,2075)
INPT191      DO 300 L=1,NUMPC
INPT192      READ(5,1060) IBC(L),JBC(L),PR(L,1),PR(L,2)
INPT193      300 WRITE(6,2080) IBC(L),JBC(L),PR(L,1),PR(L,2)
INPT194      310 CONTINUE
INPT195 C
INPT196 C      CHANGE MATERIAL TYPE FOR ORIGINAL ELEMENTS
INPT197 C
INPT198      IF(MADD.EQ.0) GO TO 336
INPT199      WRITE(6,2082)
INPT200      READ(5,1030) (MAD(I),IX(MAD(I),5),I=1,MADD)

```

```

INPT201      WRITE(6,2083) (MAD(I),IX(MAD(I),5),I=1,MAND)
INPT202      DO 335 I=1,MAND
INPT203      N=MAD(I)
INPT204      DO 335 J=1,3
INPT205      SIGIIN(J)=0.
INPT206      335 (PSIIN(J)=0.
INPT207      336 CONTINUE
INPT208 C
INPT209 C      IF INITIAL STRESS CONDITION IS TO BE ANALYSED INSTEAD OF INPUT,
INPT210 C      USE NUMBER = 0 AND DIRECTLY PROCEED TO STIFF.
INPT211 C
INPT212 C
INPT213 C      CORRECT MATERIAL TYPE FOR ELEMENTS REMOVED OR ADDED. SET INITIAL
INPT214 C      STRESSES EQUAL TO ZERO
INPT215 C
INPT216      IF (NUMBER.EQ.0) GO TO 340
INPT217      DO 330 I=1,NUMBER
INPT218      NUM = NUMR(I)
INPT219      IXINUM,5)=MTYPE
INPT220      DO 330 J=1,3
INPT221      SIGI(NUM,J) = 0.0
INPT222      330 EPSIINUM,J)=0.0
INPT223      340 SR=0.
INPT224      345 CONTINUE
INPT225      DO 350 N=1,NUMEL
INPT226      350 RATIOIN)=1.0
INPT227 C
INPT228 C      FORM STIFFNESS MATRIX IN BLOCKS FOR THIS STEP
INPT229 C
INPT230      CALL      SOLVE (R,Z,CODE,UU,CU,BI,BJ,T,SIGI,EPSI,SIGNM,BETA,
INPT231      .      RATIO,PR,AA(N15),AA(N16),IX,MTAG,NTAG,JNT,IBC,JBC)
INPT232      IF (ISTOP.EQ.1.) CALL EXIT
INPT233 C
INPT234 C      SOLVE MATRIX EQUATIONS
INPT235 C
INPT236      CALL BANSOL (AA(N15),AA(N16),MBAND,NBAND,NUMBLK,NL,JA)
INPT237      CALL      STRESS(R,Z,CODE,UU,CU,BI,BJ,T,SIGI,EPSI,SIGNM,BETA,
INPT238      .      RATIO,PR,AA(N15),AA(N16),IX,MTAG,NTAG,JNT,IBC,JBC)
INPT239 C
INPT240 C
INPT241      CALL SCALF (IX,MTAG,NTAG, JNT,RATIO,BETA,SIGI,SIGNM,EPSI,CU,
INPT242      .      AA(N15),AA(N16))
INPT243 C
INPT244 C
INPT245      JJJ=1
INPT246      IF (JCHECK.EQ.0) GO TO 400
INPT247      III=III+1
INPT248      WRITE(6,2084) LLL,III
INPT249      NKT=0
INPT250      NML=NUMEL/2

```

```

INPT251      DO 380 N=1,NUMEL
INPT252      IF(MTAG(N).GT.2) NKT=NKT+1
INPT253      IF(NKT.LT.NML) GO TO 380
INPT254      WRITE(6,2085) NKT,NML
INPT255      CALL EXIT
INPT256      380 CONTINUE
INPT257      GO TO 345
INPT258      400 IF(TOTAL.EQ.1.0 ) GO TO 600
INPT259      LLL=LLL+1
INPT260      III=0
INPT261      450 GO TO 345
INPT262      600 CONTINUE
INPT263      IF(NCASE.GE.1) GO TO 800
INPT264 C
INPT265 C      IF THE INITIAL STRESSES ARE EVALUATED IN THIS STEP
INPT266 C      1 INITIALIZE CU
INPT267 C
INPT268      DO 750 I = 1,NEQ
INPT269      750 CU(I) = 0.0
INPT270      800 NCASE = NCASE + 1
INPT271      IF(NCASE.LE.NSTEP) GO TO 200
INPT272      1010 FORMAT (2I5,1F10.3,2I5)
INPT273      RETURN
INPT274      1015 FORMAT(3F10.3)
INPT275      1020 FORMAT(F10.3,E10.0,2F10.4)
INPT276      1022 FORMAT(4F10.4)
INPT277      1025 FORMAT (15,F5.1,5F10.4)
INPT278      1030 FORMAT(16I5)
INPT279      1035 FORMAT(5I5)
INPT280      1040 FORMAT(2I5,F10.5)
INPT281      1045 FORMAT(18A4)
INPT282      1050 FORMAT(8I5)
INPT283      1055 FORMAT (16I5)
INPT284      1060 FORMAT(2I5,2F10.3)
INPT285      2010 FORMAT (17HOMATERIAL NUMBER= 13, 30H, NUMBER OF TEMPERATURE CARDS=
INPT286      1 13, 15H, MASS DENSITY= E12.4,16H, MATERIAL CODE= 13
INPT287      2,16H, MATERIAL I.D.= 13)
INPT288      2015 FORMAT(18HOTENSILE STRENGTH=E12.4,21H, COEFF. OF FRICTION=E12.4
INPT289      *,19H, INITIAL OPENING = E12.4)
INPT290      2020 FORMAT (15HO TEMPERATURE 10X 5HE      9X 6HNU      5X 8HALPHA      /
INPT291      *(F15.2,3E15.3))
INPT292      2022 FORMAT(15HO MODULUS RATIO,15H ANGLE OF FAULT,15H FRICTION COEF.,
INPT293      *15H SHEAR STRENGTH /(4E15.3))
INPT294      2025 FORMAT (108HINODAL POINT      TYPE X ORDINATE Y ORDINATE X LO
INPT295      1AD OR DISPLACEMENT Y LOAD OR DISPLACEMENT TEMPERATURE )
INPT296      2030 FORMAT (112,F12.2,2F12.5,2E24.7,F12.3)
INPT297      2035 FORMAT (26HONODAL POINT CAPD ERROR N= 15)
INPT298      2040 FORMAT(120H1ELEMENT NO.      I      J      K      L      MATERIAL
INPT299      1SIGIX      SIGIYY      SIGIXY
INPT300      2045 FORMAT (113,4I6,112,3F12.3)

```

```

INPT301 2050 FORMAT(1H0,' INITIALLY CRACKED ELEMENTS',, EL. MTAG      BITA')
INPT302 2055 FORMAT(1H1 '0A4'////)
INPT303 2060 FORMAT( ' RESULTS AFTER STEP NO.-----',15/
INPT304 1' TOTAL NUMBER OF NODAL POINTS IN SYSTEM AT THIS STEP-----',15/
INPT305 2' TOTAL NUMBER OF ELEMENTS IN SYSTEM AT THIS STEP-----',15/
INPT306 3' TOTAL NUMBER OF PRESSURE CARDS AT THIS STEP -----',15/
INPT307 4' TOTAL NUMBER OF NODAL POINTS MISSING AT THIS STEP-----',15/
INPT308 5' ELEMENTS REMOVED IN THIS STEP ARE ' /(2015))
INPT309 2065 FORMAT( ' NUMBERS OF NODAL POINTS MISSING IN THIS STEP ARE ' /
INPT310 1(2015))
INPT311 2070 FORMAT( ' RESULTS AFTER STEP NO.-----',15/
INPT312 1' TOTAL NUMBER OF NODAL POINTS IN SYSTEM AT THIS STEP-----',15/
INPT313 2' TOTAL NUMBER OF ELEMENTS IN SYSTEM AT THIS STEP-----',15/
INPT314 3' TOTAL NUMBER OF PRESSURE CARDS AT THIS STEP -----',15/
INPT315 4' TOTAL NUMBER OF NODAL POINTS MISSING AT THIS STEP-----',15/
INPT316 5' ELEMENTS ADDED IN THIS STEP ARE ' /(2015))
INPT317 2075 FORMAT(29H0PRLSSURE BOUNDARY CONDITIONS/40H 1 J PRESSUR
INPT318 1E 1 PRESSURE J)
INPT319 2080 FORMAT(216,2F14.3)
INPT320 2082 FORMAT(///' MATERIAL TYPES CHANGED AT THIS STEP'///' EL. MTYPE')
INPT321 2083 FORMAT(215//)
INPT322 2084 FORMAT(1H1,' START ITERATION TO OBTAIN SYSTEM STABILITY UNDER THE
INPT323 *LOAD INCREMENT NO.',13/
INPT324 * SYSTEM STABILITY ITERATION NO.',13//)
INPT325 2085 FORMAT(' NUMBER OF ELEMENTS CRACKED -',14,'- REACHES THE LIMIT-',1
INPT326 *4,'- PROGRESSIVE FRACTURE IS LEADING THE SYSTEM TO TOTAL FAILURE')
INPT327 2090 FORMAT (70H PROGRAM EXECUTION TERMINATED. REQUIRED CORE EXCEEDS MT
INPT328 .OUT EY 110)
INPT329 4000 FORMAT (47H FOR THIS PROGRAM THE LOCATION USED IN AA IS = 15,
INPT330 . 17H AND IN IA IS = 15/
INPT331 . 18H MAX BLOCK SIZE = 15/
INPT332 . 18H MAX BAND WIDTH = 15/ )
INPT333 4010 FORMAT (25H0 NL IS LESS THAN 2*MBAND /
INPT334 . 5H0 NL= 15/
INPT335 . 6H0 MBAND= 15)
INPT336 FND

```

```

SOLV 1      SUBROUTINE SOLVE (R,Z,CODE,UU,CU,BI,BJ,T,SIG1,EPS1,SIGNM,BETA,
SOLV 2      .   RATIO,PR,H,A,IX,MTAG,NTAG,JNT,IBC,JBC)
SOLV 3 C
SOLV 4      COMMON/ONE/ NUMNP,NUMEL,NUMMAT,NUMPC,NPC,MBAND,NUMBLK,NL,MTYPE,N,
SOLV 5      .   VOL,ACFLR,ACFLZ,Q,HFD(18),STOP,SR,SR1,TOTAL,TOL,TOL1,XC,YC,
SOLV 6      .   TEMP,SIGN,SIGDI,SIGDJ,
SOLV 7      .   LLL,III,JJJ,JCK,KCHECK,JCHECK,NPLANE,NUMR,NCODE,JA,NEQ
SOLV 8      .   ,NBAND,NCRACK,NSTEP,N15,M7,NTOT
SOLV 9      COMMON/TWO/ C(3,3),S(10,10),SIG(6),P(8),ST(3,10),RR(5),ZZ(5),
SOLV 10     .   LM(4),EE(3),EPS(3)
SOLV 11     COMMON/THREE/ F(6,4,8),EQ(4,8),TENS(8),XNU(8),RO(8),EPST(6),
SOLV 12     .   MTC(8),NTC(8),MID(8)
SOLV 13     DIMENSION R(NUMNP),Z(NUMNP),CODE(NUMNP),UU(NUMNP,2),CU(NEQ),
SOLV 14     .   ,BJ(NEQ),BJ(NEQ),T(NUMNP),SIG1(NUMEL,6),EPS1(NUMEL,6),SIGNM(NUMEL),
SOLV 15     .   ,BETA(NUMEL),RATIO(NUMEL),PR(NPC,2),IX(NUMEL,5),MTAG(NUMEL),
SOLV 16     .   ,NTAG(NUMEL),JNT(NUMEL),IBC(NPC),JBC(NPC)
SOLV 17     DIMENSION B(NL),A(NL,NBAND)
SOLV 18     NB=NL/4
SOLV 19     ND=2*NH
SOLV 20     ND2=2*ND
SOLV 21     STOP=0.0
SOLV 22     NUMBLK=D
SOLV 23     NBK=1
SOLV 24     JA=ND*(MBAND+1)/1500+1
SOLV 25     DO 50 N=1,ND2
SOLV 26     B(N)=0.0
SOLV 27     DO 50 M=1,MBAND
SOLV 28     50 A(N,M)=0.0
SOLV 29     60 NUMBLK=NUMBLK+1
SOLV 30     NH=NB*(NUMBLK+1)
SOLV 31     NM=NH-NB
SOLV 32     NLL=NM-NB+1
SOLV 33     KSHIFT=2*NLL-2
SOLV 34     DO 210 N=1,NUMEL
SOLV 35     MTAG1=MTAG(N)
SOLV 36     ICHECK=0
SOLV 37     LCHECK=1
SOLV 38     ID=N
SOLV 39     MM=4
SOLV 40     IF(IX(N,3).EQ.IX(N,4)) MM=3
SOLV 41     IF(IX(N,3).EQ.IX(N,2)) MM=2
SOLV 42     IF(IX(N,5)) 210,210,65
SOLV 43     65 DO 80 I=1,4
SOLV 44     IF(IX(N,I)-NLL) 80,70,70
SOLV 45     70 IF(IX(N,I)-NM) 90,90,80
SOLV 46     80 CONTINUE
SOLV 47     GO TO 210
SOLV 48     90 IF(NTAG(N).EQ.1) GO TO 92
SOLV 49     IF(MTAG(N).GT.2) GO TO 99
SOLV 50     IF(LLL.GT.1).AND.(MTAG(N).EQ.0) GO TO 99

```



```

SOLV 51      IF((I1).GT.0).AND.(MTAG(N).EQ.0)) GO TO 99
SOLV 52      92 IF(IX(N,3)-IX(N,2)) 95,65,95
SOLV 53      85 CALL      ONED (R,Z,CODE,UU,CU,B1,BJ,T,SIG1,EPS1,SIGNM,BETA,
SOLV 54      .      RATIO,PR,B,A,IX,MTAG,NTAG,JNT,IBC,JBC)
SOLV 55      GO TO 98
SOLV 56      95 CALL      ELEMEN(R,Z,CODE,UU,CU,B1,BJ,T,SIG1,EPS1,SIGNM,BETA,
SOLV 57      .      RATIO,PR,B,A,IX,MTAG,NTAG,JNT,IBC,JBC)
SOLV 58      NTAG(N)=0
SOLV 59      98 ICHECK=1
SOLV 60 C      *****
SOLV 61 C      * WRITE ELEMENT INFORMATION ON FILE 2 *
SOLV 62 C      *****
SOLV 63      WRITE(2*10) ((C(I1K,JJK),JJK=1,3),EE(I1K),I1K=1,3)
SOLV 64      1,((S(JJ1,KK1),KK1=1,8),JJ1=1,8),((ST(IKK,JKK),JKK=1,8),IKK=1,3)
SOLV 65      2,((RR(J11),ZZ(J11),J11=1,4),XC,YC,TEMP,VOL,MTYPE,N
SOLV 66      99 IX(N,5)=-IX(N,5)
SOLV 67      IF((LLL.GT.1).OR.(I11.GT.0)) GO TO 101
SOLV 68      IF(VOL) 100,100,101
SOLV 69      100 WRITE(6,2000) N
SOLV 70      STOP=1.0
SOLV 71 C      *****
SOLV 72 C      * CALCULATE NODAL POINT FORCES *
SOLV 73 C      *****
SOLV 74      101 IF((LLL.EQ.1).AND.(I11.EQ.0)) GO TO 145
SOLV 75      IF(ICHECK.EQ.1) GO TO 105
SOLV 76      READ(2*10) ((C(I1K,JJK),JJK=1,3),EL(I1K),I1K=1,3)
SOLV 77      1,((S(JJ1,KK1),KK1=1,8),JJ1=1,8),((ST(IKK,JKK),JKK=1,8),IKK=1,3)
SOLV 78      2,((RR(J11),ZZ(J11),J11=1,4),XC,YC,TEMP,VOL,MTYPE,N
SOLV 79      GO TO 166
SOLV 80 C
SOLV 81 C      EVALUATE STRESS TRANSFORMATION MATRIX AND STRESSES TO BE RELEASED
SOLV 82 C      FOR NEWLY FRACTURED ELEMENT ONLY
SOLV 83 C
SOLV 84      105 IF(IX(N,2).EQ.IX(N,3)) GO TO 166
SOLV 85      DO 106 I=1,3
SOLV 86      IF(SIG1(N,I).NE.0.) GO TO 110
SOLV 87      106 CONTINUE
SOLV 88      LCHECK=0
SOLV 89      GO TO 154
SOLV 90      110 IF (MTAG(N).NE.0) GO TO 111
SOLV 91      SIG(1)=0.
SOLV 92      SIG(2)=0.
SOLV 93      SIG(3)=0.
SOLV 94      GO TO 154
SOLV 95      111 IF(MTAG(N).NE.4) GO TO 112
SOLV 96      DO 112 I=1,3
SOLV 97      SIG(I)=-SIG1(N,I)
SOLV 98      112 SIG1(N,I)=0.
SOLV 99      GO TO 154
SOLV100      113 EPSX=BETA(N)

```



```

SOLV101      CC=COS(EP5X)
SOLV102      SS=SIN(EP5X)
SOLV103      S2=SS*SS
SOLV104      C2=CC*CC
SOLV105      SC=SS*CC
SOLV106      SC2=2.0*SC
SOLV107      CSD=C2-S2
SOLV108      SIGXX=C2*SIG1(N,1)+S2*SIG1(N,2)+SC2*SIG1(N,3)
SOLV109      SIGYY=S2*SIG1(N,1)+C2*SIG1(N,2)-SC2*SIG1(N,3)
SOLV110      SIGXY=-SC*(SIG1(N,1)-SIG1(N,2))+CSD*SIG1(N,3)
SOLV111      DX=EE(2)*SIGXX
SOLV112      SIG(1)=C2*SIGXX+S2*DX-SC2*SIGXY
SOLV113      SIG(2)=S2*SIGXX+C2*DX+SC2*SIGXY
SOLV114      SIG(3)=SC*(SIGXX-DX)+CSD*SIGXY
SOLV115      DO 144 I=1,3
SOLV116      SIG1(N,I)=SIG1(N,I)-SIG(I)
SOLV117      144 SIG(I)=-SIG(I)
SOLV118      GO TO 154
SOLV119      C
SOLV120      C      CALCULATE TEMPERATURE STRESSES ,FOR LLL=1 AND III=0 ONLY
SOLV121      C
SOLV122      145 DT=TEMP-Q
SOLV123      DX=EE(3)*DT
SOLV124      SIG(1)=-(C(1,1)+C(1,2))*DX+SIG1(N,1)
SOLV125      SIG(2)=-(C(2,2)+C(1,2))*DX+SIG1(N,2)
SOLV126      SIG(3)=SIG1(N,3)
SOLV127      154 IF(IX(N,2).EQ.IX(N,3)) GO TO 166
SOLV128      DO 160 I=1,8
SOLV129      P(I)=0.0
SOLV130      IF(LCHECK.EQ.0) GO TO 160
SOLV131      DO 155 J=1,3
SOLV132      155 P(I)=P(I)-ST(J,I)*SIG(J)
SOLV133      160 P(I)=P(I)*VOL
SOLV134      IF((LLL.GT.1).OR.(III.GT.0)) GO TO 166
SOLV135      C
SOLV136      C      CALCULATE BODY FORCES, FOR LLL=1 AND III=0 ONLY
SOLV137      C
SOLV138      163 XMM=MM
SOLV139      DY=VOL*ACFLZ*RU(MTYPE)/XMM
SOLV140      DX=VOL*ACELR*RU(MTYPE)/XMM
SOLV141      DO 165 I=1,MM
SOLV142      P(2*I)=P(2*I)+DY
SOLV143      165 P(2*I-1)=P(2*I-1)+DX
SOLV144      166 CONTINUE
SOLV145      167 DO 168 I=1,MM
SOLV146      168 LM(I)=2*IX(N,I)-2
SOLV147      DO 200 I=1,MM
SOLV148      DO 200 K=1,2
SOLV149      II=LM(I)+K-KSHIFT
SOLV150      KK=2*I-2+K

```

```

SOLV151      IF (ICHECK.EQ.0) GO TO 172
SOLV152      B(11)=B(11)+P(KK)
SOLV153      172 DO 200 J=1,MM
SOLV154      DO 200 L=1,2
SOLV155      JJ=LM(J)+L-11+1-KSHIFT
SOLV156      LL=2*J-L
SOLV157      IF (JJ) 200,200,175
SOLV158      175 IF (ND-JJ) 180,195,195
SOLV159      180 WRITE (6,2001) N
SOLV160      STOP=1.0
SOLV161      GO TO 210
SOLV162      195 A(11,JJ)=A(11,JJ)+S(KK,LL)
SOLV163      200 CONTINUE
SOLV164      210 CONTINUE
SOLV165      IF (LLL.GT.1).OR.(111.GT.0)) GO TO 301
SOLV166      DO 220 N=NLL,NM
SOLV167      IF (N-NUMNP) 215,215,220
SOLV168      215 K=2*N-KSHIFT
SOLV169      B(K)=B(K)+UU(N,2)
SOLV170      B(K-1)=B(K-1)+UU(N,1)
SOLV171      220 CONTINUE
SOLV172      IF (NUMPC) 225,301,225
SOLV173      225 DO 300 L=1,NUMPC
SOLV174      I=180(L)
SOLV175      J=J60(L)
SOLV176      DR=Z(1)-Z(J)
SOLV177      DZ=R(J)-R(1)
SOLV178      PP2=(PR(L,2)+PR(L,1))/6.
SOLV179      PP1=PP2+PR(L,1)/6.
SOLV180      PP2=PP2+PR(L,2)/6.
SOLV181      11=2*I-KSHIFT
SOLV182      JJ=2*J-KSHIFT
SOLV183      IF (11) 265,265,235
SOLV184      235 IF (11-ND) 240,240,265
SOLV185      240 B(11-1)=B(11-1)+PP1*DR
SOLV186      B(11)=B(11)+PP1*DZ
SOLV187      265 IF (JJ) 300,300,270
SOLV188      270 IF (JJ-ND) 275,275,300
SOLV189      275 B(JJ-1)=B(JJ-1)+PP2*DR
SOLV190      B(JJ)=B(JJ)+PP2*DZ
SOLV191      300 CONTINUE
SOLV192      301 CONTINUE
SOLV193      11=ND*(NUMBLK-1)+1
SOLV194      JJ=ND*NUMBLK
SOLV195      KK=C
SOLV196      IF (111.NE.0) GO TO 303
SOLV197      DO 302 N=11,JJ
SOLV198      302 BJ(N)=0.
SOLV199      303 IF (LLL.EQ.1).AND.(111.EQ.0) GO TO 305
SOLV200      GO TO 313

```

```

SOLV201 305 DO 306 N=11,JJ
SOLV202 KK=KK+1
SOLV203 306 R1(N)=B(KK)
SOLV204 GO TO 310
SOLV205 J13 FACTOR=SR
SOLV206 IF (I1.EQ.0) FACTOR=1.-TOTAL
SOLV207 IF (LL1.EQ.1) FACTOR=TOTAL
SOLV208 DO 314 N=11,JJ
SOLV209 KK=KK+1
SOLV210 BJ(N)=BJ(N)+B(KK)
SOLV211 314 B(KK)=FACTOR*B1(N)+BJ(N)
SOLV212 310 DO 400 M=NLL,NH
SOLV213 IF (M-NUMNP) 315,315,400
SOLV214 315 U=UU(M,1)
SOLV215 N=2*M-1-KSHIFT
SOLV216 IF (CODE(M)) 390,400,316
SOLV217 316 IF (CODE(M)-1.) 317,370,317
SOLV218 317 IF (CODE(M)-2.) 318,390,316
SOLV219 318 IF (CODE(M)-3.) 390,380,390
SOLV220 370 CALL MODIFY(A,B,ND2,MBAND,NBAND,N,U)
SOLV221 GO TO 400
SOLV222 380 CALL MODIFY(A,B,ND2,MBAND,NBAND,N,U)
SOLV223 390 U=UU(M,2)
SOLV224 N=N+1
SOLV225 CALL MODIFY(A,E,NU2,MBAND,NBAND,N,U)
SOLV226 400 CONTINUE
SOLV227 C *****
SOLV228 C * WRITE BLOCK INFORMATION ON FILE 1 *
SOLV229 C *****
SOLV230 WRITE(1,NBK)(B(N),(A(N,M),M=1,MBAND),N=1,ND)
SOLV231 NBK=NBK+JA
SOLV232 DO 420 N=1,ND
SOLV233 K=N+ND
SOLV234 B(N)=B(K)
SOLV235 B(K)=0.0
SOLV236 DO 420 M=1,MBAND
SOLV237 A(N,M)=A(K,M)
SOLV238 420 A(K,M)=0.0
SOLV239 IF (NH-NUMNP) 60,460,480
SOLV240 480 CONTINUE
SOLV241 RETURN
SOLV242 2000 FORMAT (26HNEGATIVE AREA ELEMENT NO. 14)
SOLV243 2001 FORMAT (29HOBAND WIDTH EXCEEDS ALLOWABLE 14)
SOLV244 END

```

```

UNED 1  SUBROUTINE UNED (R,Z,CIDE,UU,CU,BI,BJ,T,SIG,EPCI,SICNM,BETA,
UNED 2  . PATIC,PR,D,A,IX,MTAG,NTAG,JNT,IBC,JBC)
UNED 3  COMMON/UNF/ NUMNP,NUMEL,NUMMAT,NUMPC,NPC,NBAND,NUMBLK,NL,MTYPE,N,
UNED 4  . VOL,ACELR,ACELZ,Q,HED(18),STUP,JR,SR1,TOTAL,TOL,TUL1,XC,YC,
UNED 5  . TEMP,SIGN,SIGDI,SIGDJ,
UNED 6  . LLL,TII,JJJ,JCK,KCHECK,JCHECK,NPLANE,NUMFR,NCODE,JA,NEQ
UNED 7  . NBAND,NCRACK,NSTEP,N15,N7,NTOT
UNED 8  COMMON/TWO/ C(2,3),S(10,10),SIG(6),P(8),ST(3,10),RR(5),ZZ(5),
UNED 9  . LM(4),FE(3),EPS(3)
UNED 10 COMMON/THREE/ I(8,4,8),LU(4,8),TENS(8),XNU(8),RU(8),EPST(8),
UNED 11 . MTC(8),NTC(8),MID(8)
UNED 12 DIMENSION R(NUMNP),Z(NUMNP),CODE(NUMNP),UU(NUMNP,2),CU(NEQ),
UNED 13 .BI(NEQ),BJ(NEQ),T(NUMNP),SIGI(NUMEL,6),EPSI(NUMEL,6),SIGNM(NUMEL),
UNED 14 .BETA(NUMEL),RATIO(NUMEL),PR(NPC,2),IX(NUMEL,5),MTAG(NUMEL),
UNED 15 .NTAG(NUMEL),JNT(NUMEL),IBC(NPC),JBC(NPC)
UNED 16 DIMENSION B(NL),A(NL,NBAND)
UNED 17 C
UNED 18 DO 100 I=1,8
UNED 19 P(I)=0.0
UNED 20 DO 100 J=1,8
UNED 21 100 S(I,J)=0.0
UNED 22 MTYPE=IX(N,5)
UNED 23 I=IX(N,1)
UNED 24 J=IX(N,2)
UNED 25 DX=R(J)-R(I)
UNED 26 DY=Z(J)-Z(I)
UNED 27 XL= SQRT(DX**2+DY**2)
UNED 28 COSA=DX/XL
UNED 29 SINA=DY/XL
UNED 30 EF(1)=E(1,2,MTYPE)
UNED 31 EE(2)=E(1,3,MTYPE)
UNED 32 COMM=EE(1)*EE(2)/XL
UNED 33 S(1,1)=COSA*COSA*COMM
UNED 34 S(1,2)=COSA*SINA*COMM
UNED 35 S(1,3)=-S(1,1)
UNED 36 S(1,4)=-S(1,2)
UNED 37 S(2,1)=S(1,3)
UNED 38 S(2,2)=SINA*SINA*COMM
UNED 39 S(2,3)=-S(1,2)
UNED 40 S(2,4)=-S(2,2)
UNED 41 S(3,1)=S(1,3)
UNED 42 S(3,2)=S(2,3)
UNED 43 S(3,3)=S(1,1)
UNED 44 S(3,4)=S(1,2)
UNED 45 S(4,1)=S(1,4)
UNED 46 S(4,2)=S(2,4)
UNED 47 S(4,3)=S(3,4)
UNED 48 S(4,4)=S(2,2)
UNED 49 EP=E(1,4,MTYPE)*E(1,1,MTYPE)
UNED 50 DX=DX*EP

```

```

UNED 51      DY=DY*P
UNED 52      P(1)=S(1,1)*D(+S(1,2)*DY
UNED 53      P(2)=S(2,1)*X+S(2,2)*DY
UNED 54      136 P(3)=-P(1)
UNED 55      P(4)=-P(2)
UNED 56      VOL=1.
UNED 57      DO 140 I=1,4
UNED 58      NPP=IX(N,I)
UNED 59      RR(I)=R(NPP)
UNED 60      140 ZZ(1)=Z(NPP)
UNED 61      XC=0.5*(RR(1)+RR(2))
UNED 62      YC=0.5*(ZZ(1)+ZZ(2))
UNED 63      DO 200 I=1,3
UNED 64      DO 200 J=1,8
UNED 65      200 ST(I,J)=0.0
UNED 66      DO 300 I=1,3
UNED 67      DO 300 J=1,3
UNED 68      300 C(I,J)=0.0
UNED 69      DO 400 I=1,3
UNED 70      400 EE(I)=L(1,I+1,MTYPE)
UNED 71 C
UNED 72      RETURN
UNED 73 C
UNED 74      END

```

```

ELEM 1  SUBROUTINE CLEMEM(R,Z,COEF,UU,CU,BI,BJ,T,SIG,FPSI,SIGNM,BETA,
ELEM 2  . RAT,0,PR,F,A,IX,MTAG,NTAG,JNT,IBC,JBC)
ELEM 3  COMMON/ONE/ NUMNP,NUMEL,NUMMAT,NUMPC,NPC,MEAND,NUMBLK,NL,MTYPE,N,
ELEM 4  . VOL,ACFLK,ACELZ,Q,HED(18),STUP,SR,SR1,TOTAL,TOL,TOL1,XC,YC,
ELEM 5  . TEMP,SIGN,SIGD1,SIGD2,
ELEM 6  . LLL,111,JJJ,JCK,KCHLCK,JCHECK,NPLANE,NUMR,NCODE,JA,NED
ELEM 7  . NBAND,NCHALK,NSTEP,N15,M7,NTOT
ELEM 8  COMMON/TWO/ C(3,3),S(10,10),SIG(6),P(6),ST(3,10),RP(5),ZZ(5),
ELEM 9  . LM(4),FF(3),EPS(3)
ELEM 10 COMMON/THREE/ E(6,4,8),FO(4,8),IENS(8),XNU(8),RO(8),EPST(8),
ELEM 11 . MIC(8),NTC(8),MID(8)
ELEM 12 DIMENSION K(NUMNP),Z(NUMNP),CODE(NUMNP),UU(NUMNP,2),CU(NUMNP),
ELEM 13 . BI(NUMNP),BJ(NUMNP),T(NUMNP),SIG(NUMEL,6),FPSI(NUMEL,6),SIGNM(NUMEL),
ELEM 14 . BETA(NUMEL),RATIO(NUMEL),PR(NPC,2),IX(NUMEL,5),MTAG(NUMEL),
ELEM 15 . NIAG(NUMEL),JNT(NUMEL),IBC(NPC),JBC(NPC)
ELEM 16 DIMENSION R(NL),A(NL,NBAND)
ELEM 17 DIMENSION U(3),V(3)
ELEM 18 C
ELEM 19 .FX=0.00001
ELEM 20 I=IX(N,1)
ELEM 21 J=IX(N,2)
ELEM 22 K=IX(N,3)
ELEM 23 L=IX(N,4)
ELEM 24 MTYPE=IX(N,5)
ELEM 25 VOL=0.
ELEM 26 TEMP=(T(I)+T(J)+T(K)+T(L))/4.0
ELEM 27 RAT=0.0
ELEM 28 NUMTC=NTC(MTYPE)
ELEM 29 IF (NUMTC.EQ.1) GO TO 100
ELEM 30 DO 50 M=2,NUMTC
ELEM 31 IF (E(M,1,MTYPE)-TEMP) 50,60,60
ELEM 32 50 CONTINUE
ELEM 33 60 DEN=E(M,1,MTYPE)-E(M-1,1,MTYPE)
ELEM 34 IF (DEN) 70,80,70
ELEM 35 70 RAT = (TEMP-E(M-1,1,MTYPE))/DEN
ELEM 36 80 DO 90 KK=1,3
ELEM 37 90 FF(KK)=E(M-1,KK+1,MTYPE)+RAT *(E(M,KK+1,MTYPE)-E(M-1,KK+1,MTYPE))
ELEM 38 GO TO 100
ELEM 39 100 DO 105 KK=1,3
ELEM 40 105 FF(KK)=E(1,KK+1,MTYPE)
ELEM 41 106 CONTINUE
ELEM 42 C *****
ELEM 43 C * CALCULATE ORTHOTROPIC ELASTICITY MATRIX FOR MTAG=0,1,2 *
ELEM 44 C *****
ELEM 45 KKK=MTAG(N)+1
ELEM 46 GO TO (117,107,108),KKK
ELEM 47 107 CONTINUE
ELEM 48 C
ELEM 49 C NIAG=1
ELEM 50 C

```

```

ELEM 51      EX=EX
ELEM 52      EY=1.
ELEM 53      GO TO 109
ELEM 54      108 CONTINUE
ELEM 55 C
ELEM 56 C      MTAG=2
ELEM 57 C
ELEM 58      EX=EX
ELEM 59      EY=EX
ELEM 60      109 CONTINUE
ELEM 61      GO TO (110,111),NPLANE
ELEM 62      110 CONTINUE
ELEM 63 C
ELEM 64 C      PLANE STRESS
ELEM 65 C
ELEM 66      EE1=EE(1)*EX
ELEM 67      EE2=EE(1)*EY
ELEM 68      EFN=EE(2)*EY
ELEM 69      COMM=(1.-(EE1/EE2)/(EEN**2))
ELEM 70      C11=EE1/COMM
ELEM 71      C12=EEN*C11
ELEM 72      C21=C12
ELEM 73      C22=EE2/COMM
ELEM 74      GO TO 115
ELEM 75      111 CONTINUE
ELEM 76 C
ELEM 77 C      PLANE STRAIN
ELEM 78 C
ELEM 79      EE1=EE(1)*EX
ELEM 80      EE2=EE(1)*EY
ELEM 81      EN1=EE(2)*EX
ELEM 82      EN2=EE(2)*EY
ELEM 83      EN3=EE(2)
ELEM 84      XA=1.-EN1*EN3
ELEM 85      XB=1.-EN2*EN3
ELEM 86      XC=1.+EN3
ELEM 87      XD=EN2*XC
ELEM 88      XE=EE2/EE1
ELEM 89      COMM=(XE*XA*XB-(EN2**2)*(XC**2))/EE2
ELEM 90      C11=XB/COMM
ELEM 91      C12=XD/COMM
ELEM 92      C21=C12
ELEM 93      C22=XE*XA/COMM
ELEM 94      115 CONTINUE
ELEM 95      EPSX=BETA(N)
ELEM 96      KCK=0
ELEM 97      IF(MID(MTYPE).EQ.0) GO TO 112
ELEM 98      EPSX=EO(2,MTYPE)
ELEM 99      IF(MTAG(N).NE.0) EPSX=EPSX-BETA(N)
ELEM100      112 CONTINUE

```



```

ELEM101      S1= SIN(EPSX)
ELEM102      S1=S1*S1
ELEM103      C1= COS(EPSX)
ELEM104      C2=C1*C1
ELEM105      C1S1=C1*S1
ELEM106      C2S2=C2*S2
ELEM107      C4=C2**2
ELEM108      S4=S2*S2
ELEM109      DA=C11+C22
ELEM110      DB=C11-C12
ELEM111      DC=C22-C12
ELEM112      DD=2.*C2S2*C1
ELEM113      C(1,1)=C4*C11+DD+S4*C22
ELEM114      C(1,2)=C2S2*DA+(C4+S4)*C12
ELEM115      C(1,3)=C1S1*(C2*DB-S2*DC)
ELEM116      C(2,2)=S4*C11+DD+C4*C22
ELEM117      C(2,3)=C1S1*(S2*DB-C2*DC)
ELEM118      C(3,3)=C2S2*(DB+DC)
ELEM119      IF((MID(MTYPE).EQ.0).OR.(KCK.EQ.1)) GO TO 116
ELEM120      IF(NTAG(N).NE.0) KCK=1
ELEM121      G=0.5*EE1/(1.+(EX**2)*EE2)
ELEM122      EPSX=2.*I*PSX
ELEM123      S1=SIN(EPSX)
ELEM124      C1=COS(EPSX)
ELEM125      SC=S1*C1
ELEM126      S1=S1*S1
ELEM127      C1=C1*C1
ELEM128      C11=G*S1
ELEM129      C12=-C11
ELEM130      C13=-G*SC
ELEM131      C22=C11
ELEM132      C23=-C13
ELEM133      C24=G*C1
ELEM134      C(1,1)=C(1,1)+C11
ELEM135      C(1,2)=C(1,2)+C12
ELEM136      C(1,3)=C(1,3)+C13
ELEM137      C(2,2)=C(2,2)+C22
ELEM138      C(2,3)=C(2,3)+C23
ELEM139      C(3,3)=C(3,3)+C33
ELEM140      IF(KCK.EQ.0) GO TO 116
ELEM141      EX=EE*X
ELEM142      LY=1.
ELEM143      IF(MTAG(N).EQ.1) GO TO 115
ELEM144      EY=EE*Y
ELEM145      115 CONTINUE
ELEM146      C11=EX*C(1,1)
ELEM147      C12=EX*C(1,2)
ELEM148      C24=LY*C(2,4)
ELEM149      C21=C12
ELEM150      EPSX=BETA(N)

```

```

ELEM151      GO TO 112
ELEM152      116 CONTINUE
ELEM153      C(2,1)=C(1,2)
ELEM154      C(3,2)=C(2,3)
ELEM155      C(3,1)=C(1,3)
ELEM156      IF(MTAG(N).EQ.0) GO TO 118
ELEM157      MTAG(N)=MTAG(N)+2
ELEM158      GO TO 118
ELEM159      117 CONTINUE
ELEM160      IF((MID(MTYPE).EQ.0).OR.(EQ(1,MTYPE).EQ.1.)) GO TO 125
ELEM161      EX=EQ(1,MTYPE)
ELEM162      EY=1.0
ELEM163      GO TO 109
ELEM164      125 CONTINUE
ELEM165 C      *****
ELEM166 C      * CALCULATE ISOTROPIC ELASTICITY MATRIX *
ELEM167 C      *****
ELEM168      EE1=EE(1)
ELEM169      EE2=EE(2)
ELEM170      IF(NPLANE.EQ.1) GO TO 119
ELEM171      EE1=EE1/(1.-EE2**2)
ELEM172      EE2=EE2/(1.-EE2)
ELEM173      119 CONTINUE
ELEM174      COMM=EE1/(1.-EE2**2)
ELEM175      C(1,1)=COMM
ELEM176      C(1,2)=COMM*EE2
ELEM177      C(1,3)=0.
ELEM178      C(2,1)=C(1,2)
ELEM179      C(2,2)=C(1,1)
ELEM180      C(2,3)=0.
ELEM181      C(3,1)=0.
ELEM182      C(3,2)=0.
ELEM183      C(3,3)=.5*COMM*(1.-EE2)
ELEM184      118 CONTINUE
ELEM185      DO 130 J=1,10
ELEM186      DO 120 I=1,3
ELEM187      120 ST(I,J)=0.
ELEM188      DO 130 I=1,10
ELEM189      130 S(I,J)=0.
ELEM190      135 CONTINUE
ELEM191      DO 140 I=1,4
ELEM192      NPP=IX(N,I)
ELEM193      RR(I)=R(NPP)
ELEM194      140 ZZ(I)=Z(NPP)
ELEM195      IF(IX(N,3)-IX(N,4)) 145,150,145
ELEM196      145 XC=(RR(1)+RR(2)+RR(3)+RR(4))/4.
ELEM197      YC=(ZZ(1)+ZZ(2)+ZZ(3)+ZZ(4))/4.
ELEM198      RK(5)=XC
ELEM199      ZZ(5)=YC
ELEM200      K=5

```

```

ELEM201      J=1
ELEM202      I=4
ELEM203      LM(3)=9
ELEM204      NT=4
ELEM205      GO 10 160
ELEM206      150 NT=1
ELEM207      LM(3)=5
ELEM208      I=1
ELEM209      J=2
ELEM210      K=3
ELEM211      XC=(RR(1)+RR(2)+RR(3))/3.
ELEM212      YC=(ZZ(1)+ZZ(2)+ZZ(3))/3.
ELEM213      RR(5)=RR(3)
ELEM214      ZZ(5)=ZZ(3)
ELEM215      160 DO 200 NN=1,NT
ELEM216      LM(1)=2*I-1
ELEM217      LM(2)=2*J-1
ELEM218      U(1)=ZZ(J)-ZZ(K)
ELEM219      U(2)=ZZ(K)-ZZ(1)
ELEM220      U(3)=ZZ(1)-ZZ(J)
ELEM221      V(1)=RR(K)-RR(J)
ELEM222      V(2)=RR(1)-RR(K)
ELEM223      V(3)=RR(J)-RR(1)
ELEM224      AREA=(RR(J)*U(2)+RR(1)*U(1)+RR(5)*U(3))/2.
ELEM225      VOL=VOL+AREA
ELEM226      COMM=.25/AREA
ELEM227      XNT=NT
ELEM228      COM=2./XNT
ELEM229      CUM=COM*COMM
ELEM230      DO 180 L=1,3
ELEM231      II=LM(L)
ELEM232      ST(1,II)=ST(1,II)+U(L)*COM
ELEM233      ST(2,II+1)=ST(2,II+1)+V(L)*COM
ELEM234      ST(3,II)=ST(3,II)+V(L)*COM
ELEM235      ST(3,II+1)=ST(3,II+1)+U(L)*COM
ELEM236      DO 180 M=1,3
ELEM237      JJ=LM(M)
ELEM238      S(11,JJ)=S(11,JJ)+(C(1,1)*U(L)*U(M)+C(1,3)*(V(L)*U(M)+U(L)*V(M))
ELEM239      * +C(3,3)*V(L)*V(M))*COMM
ELEM240      S(11,JJ+1)=S(11,JJ+1)+(U(L)*(C(1,2)*V(M)+C(1,3)*U(M))+V(L)*(C(2,1)
ELEM241      * *V(M)+C(3,2)*U(M))*COMM
ELEM242      S(11+1,JJ+1)=S(11+1,JJ+1)+(C(2,2)*V(L)*V(M)+C(2,3)*(U(L)*V(M)+V(L)
ELEM243      * *U(M))+C(3,3)*U(L)*U(M))*COMM
ELEM244      S(JJ+1,11)=S(11,JJ+1)
ELEM245      180 CONTINUE
ELEM246      I=J
ELEM247      J=J+1
ELEM248      200 CONTINUE
ELEM249      IF((IX(N,3)-IX(N,4)) 220,250,220
ELEM250      220 DO 240 I=1,2

```

```

FLEM251      KK=10-I
FLEM252      DO 240 K=1, KK
FLEM253      IF(S(KK+1, KK+1).EQ.0.) S(KK+1, KK+1)=.000001
FLEM254      CC=S(KK+1, K)/S(KK+1, KK+1)
FLEM255      DO 230 J=1, 3
FLEM256      230 ST(J, K)=ST(J, K)-CC*ST(J, KK+1)
FLEM257      DO 240 J=1, KK
FLEM258      240 S(J, K)=S(J, K)-CC*S(J, KK+1)
FLEM259      250 CONTINUE
FLEM260      RETURN
FLEM261 C
FLEM262      END

```

```

STRS 1      SUBROUTINE STRESS(R,Z,CODE,UU,CU,B1,BJ,T,SIG1,EPS1,SIGNM,BETA,
STRS 2      . RATIO,PR,H,A,IX,MTAG,NTAG,JNT,IBC,JBC)
STRS 3      COMMON/UNI/ NUMNP,NUMEL,NUMMAT,NUMPC,NPC,MBAND,NUMBLK,NL,MTYPE,N,
STRS 4      . VOL,ACELR,ACELZ,Q,HED(1E),STOP,SR,SK1,TOTAL,TOL,TOLI,XC,YC,
STRS 5      . TEMP,SIGN,SIGD1,SIGD2,
STRS 6      . LLE,111,JJJ,JCK,KCHECK,JCHECK,NPLANE,NUMER,NCODE,JA,NEO
STRS 7      . ,NBAND,NCRACK,NSTEP,N15,M7,NTOT
STRS 8      COMMON/TWO/ C(3,3),S(10,10),SIG(6),P(8),ST(3,10),RR(5),ZZ(5),
STRS 9      . LM(4),FE(3),EPS(3)
STRS 10     COMMON/THREE/ E(8,4,8),ED(4,8),TENS(8),XNU(8),RO(8),FPST(8),
STRS 11     . MTC(8),NLC(8),MID(8)
STRS 12     DIMENSION R(NUMNP),Z(NUMNP),CODE(NUMNP),UU(NUMNP,2),CU(NEQ),
STRS 13     .B1(NEQ),BJ(NLQ),T(NUMNP),SIG1(NUMEL,6),EPS1(NUMEL,6),SIGNM(NUMEL),
STRS 14     .BETA(NUMEL),RATIO(NUMEL),PR(NPC,2),IX(NUMEL,5),MTAG(NUMEL),
STRS 15     .NTAG(NUMEL),JNT(NUMEL),IBC(NPC),JBC(NPC)
STRS 16     DIMENSION B(NL),A(NL,NBAND)
STRS 17     DO 600 M=1,NUMEL
STRS 18     ID=M
STRS 19     FIND(2*ID)
STRS 20     IX(M,5)=IABS(IX(M,5))
STRS 21     DO 50 I=1,6
STRS 22     50 SIG(I)=0.0
STRS 23 C     *****
STRS 24 C     * READ ELEMENT INFORMATION FROM FILE 2
STRS 25 C     *****
STRS 26     READ(2*ID) ((C(I1K,JJK),JJK=1,3),EE(I1K),I1K=1,3)
STRS 27     1,((S(JJI,KKI),KKI=1,8),JJI=1,8),((ST(IKK,JKK),JKK=1,8),IKK=1,2)
STRS 28     2,((RR(J11),ZZ(J11),J11=1,4),XC,YC,TEMP,VOL,MTYPE,N
STRS 29     IF(IX(N,3)-IX(N,2)) 90,60,90
STRS 30 C     *****
STRS 31 C     * ONE-DIMENSIONAL ELEMENT
STRS 32 C     *****
STRS 33 C     *****
STRS 34     60 IF(MTAG(N).EQ.4) GO TO 70
STRS 35     I=IX(N,1)
STRS 36     J=IX(N,2)
STRS 37     DX=RR(2)-RR(1)
STRS 38     DY=ZZ(2)-ZZ(1)
STRS 39     XL= SQRT(DX**2+DY**2)
STRS 40     DU=B(2*J)-B(2*I)
STRS 41     DV=B(2*J)-B(2*I)
STRS 42     DL=DV*DY/XL+DU*DX/XL
STRS 43     SIG(1)=DL*EE(1)/XL
STRS 44     IF(EE(2).EQ.0.) SIG(1)=0.
STRS 45     70 SIG1(N,4)=SIG(1)
STRS 46     SIG1(N,5)=0.
STRS 47     SIG1(N,6)=0.
STRS 48     GO TO 600
STRS 49 C
STRS 50 C     *****

```

```

STRS 51 C      *      TWO-DIMENSIONAL ELEMENT      *
STRS 52 C      *****
STRS 53      90 MM=4
STRS 54      IF(IX(N,3).EQ.IX(N,4)) MM=3
STRS 55      170 DO 180 I=1,3
STRS 56      RR(I)=0.
STRS 57      DO 180 J=1,MM
STRS 58      II=2*J
STRS 59      JJ=2*IX(N,J)
STRS 60      180 RR(I)=RR(I)+ST(I,II)*B(JJ)+ST(I,II-1)*B(JJ-1)
STRS 61      DO 190 I=1,3
STRS 62      DO 190 J=1,3
STRS 63      SIG(I)=SIG(I)+C(I,J)*RR(J)
STRS 64      IF(MTAG(N).EQ.4) SIG(I)=0.
STRS 65      190 CONTINUE
STRS 66      IF(MTAG(N).NE.3) GO TO 191
STRS 67      EPSX=BETA(N)
STRS 68      CC=COS(EPSX)
STRS 69      SS=SIN(EPSX)
STRS 70      S2=SS*SS
STRS 71      C2=CC*CC
STRS 72      SC=SS*CC
STRS 73      SC2=2.*SC
STRS 74      CSD=C2-S2
STRS 75      SIGXX=C2*SIG(1)+S2*SIG(2)+SC2*SIG(3)
STRS 76      SIGYY=S2*SIG(1)+C2*SIG(2)-SC2*SIG(3)
STRS 77      SIGXY=-SC*(SIG(1)-SIG(2))+CSD*SIG(3)
STRS 78      DX=FF(2)*SIGXX
STRS 79      SIG(1)=SIG(1)-(C2*SIGXX+S2*DX-SC2*SIGXY)
STRS 80      SIG(2)=SIG(2)-(S2*SIGXX+C2*DX+SC2*SIGXY)
STRS 81      SIG(3)=SIG(3)-(SC*(SIGXX-DX)+CSD*SIGXY)
STRS 82      191 CONTINUE
STRS 83      DO 192 I=1,3
STRS 84      SIG(I(N,I+3))=SIG(I)
STRS 85      EPSI(N,I+3)=RR(I)
STRS 86      192 CONTINUE
STRS 87      600 CONTINUE
STRS 88      RETURN
STRS 89      END

```

```

GRIF 1      SUBROUTINE GRIFTH (MTAG,JNT,RATIO,BETA,SIGI,SIGNM)
GRIF 2 C
GRIF 3      COMMON/ONE/ NUMNP,NUMEL,NUMMAT,NUMPC,NPC,MEAND,NUMBLK,NL,MTYPE,N,
GRIF 4      . VOL,AC1LR,AC1LZ,Q,HED(18),STOP,SR,SK1,TOTAL,TOL1,XC,YC,
GRIF 5      . TEMP,SIGN,SIGD1,SIGD2,
GRIF 6      . LLL,111,JJJ,JCK,KCHECK,JCHECK,NPLANE,NUMLR,NCODE,JA,NLQ
GRIF 7      . ,NUAND,NCRACK,NSTEP,N15,M7,NTOT
GRIF 8      COMMON/TWO/ C(3,3),S(16,16),SIG(6),P(81,S1(3,101),RR(5),ZZ(5),
GRIF 9      . LM(4),EE(3),FPS(3)
GRIF 10     COMMON/THREE/ E(8,4,81,FU(4,81,TENS(81,XNU(8),RD(8),EPST(8),
GRIF 11     . MTC(8),NIC(8),MID(8)
GRIF 12     DIMENSION MTAG(NUMEL),JNT(NUMEL),RATIO(NUMEL),BETA(NUMEL),
GRIF 13     . SIGI(NUMEL,6),SIGNM(NUMEL)
GRIF 14 C      *****
GRIF 15 C      * XNU,XN= COEF. OF FRICTION FOR THE MODIFIED GRIFFITH CRITERION *
GRIF 16 C      * TENS,SIG= TENSILE STRENGTH OF THE MATERIAL *
GRIF 17 C      * SIGN= STRESS COMPONENT NORMAL TO THE FRACTURE PLANE *
GRIF 18 C      * FPSX= ANGLE BETWEEN P AND SIGN *
GRIF 19 C      * ELTA= ANGLE BETWEEN X-AXIS AND SIGN *
GRIF 20 C      * ANGLES ARE POSITIVE COUNTERCLOCKWISE *
GRIF 21 C      *****
GRIF 22     MTAG1=MTAG(N)
GRIF 23     JNT1=JNT(N)
GRIF 24     SIGYY=0.
GRIF 25     SIGN=0.
GRIF 26     IF(MTAG(N).GT.2) GO TO 600
GRIF 27     MTAG(N)=0
GRIF 28     JNT(N)=0
GRIF 29     BETA(N)=0.
GRIF 30     PP=SIG(4)
GRIF 31     Q=SIG(5)
GRIF 32     R=SIG(6)
GRIF 33     SIGT=TENS(MTYPE)
GRIF 34     XN=XNU(MTYPE)
GRIF 35     SUM=3.0*PP+Q
GRIF 36     TOL=0.005*SIGT
GRIF 37     TOLL=TOL
GRIF 38     IF(JCK.EQ.0) TOLL=0.
GRIF 39     MD=MID(MTYPE)
GRIF 40     IF((MTAG1.EQ.2).AND.(JNT1.LT.411) GO TO 30
GRIF 41 C      CHECK INITIAL WEAK PLANE
GRIF 42     IF(MD.EQ.0) GO TO 30
GRIF 43     WX=2.*(EL(2,MTYPE1-R/57.296)
GRIF 44     W=0.5*WX
GRIF 45     X=COS(WX)
GRIF 46     PQADD=PP+Q
GRIF 47     PQSUB=PP-Q
GRIF 48     SIGYY=0.5*(PQADD+PQSUB*X)
GRIF 49     SIGI=F0(4,MTYPE1
GRIF 50     XN=EO(3,MTYPE)

```



```

GRIF 51      IF(SIGYY.LE.0.) GO TO 200
GRIF 52      SIGN=SIGYY
GRIF 53      GO TO 40
GRIF 54      30 CONTINUE
GRIF 55      IF(SUM.LT.0.) GO TO 100
GRIF 56      SIGN=PP
GRIF 57      40 CONTINUE
GRIF 58      DSIG=SIGN-SIGI
GRIF 59      IF((JCK.EQ.1).AND.(ABS(DSIG).LT.TOLL)) GO TO 50
GRIF 60      IF(DSIG.LT.TOLL) GO TO 600
GRIF 61 C     *****
GRIF 62 C     * FRACTURE PLANE COINCIDES WITH SIG(5) IF CRITICAL ANGLE IS ZERO *
GRIF 63 C     * APPLY ORIGINAL GRIFFITH CRITERION *
GRIF 64 C     *****
GRIF 65      50 CONTINUE
GRIF 66      IF(MD.EQ.0) GO TO 70
GRIF 67      IF((MTAG1.EQ.2).AND.(JNT1.LT.4)) GO TO 70
GRIF 68      MTAG(N)=1
GRIF 69      JNT(N)=4
GRIF 70      BETA(N)=ED(2,MTYPE)
GRIF 71      GO TO 320
GRIF 72      70 JNT(N)=1
GRIF 73      SIGN=PP
GRIF 74      SIGYY=SIGN
GRIF 75      EPSX=0.0
GRIF 76      GO TO 300
GRIF 77      100 CONTINUE
GRIF 78 C     *****
GRIF 79 C     * CRITICAL ANGLE IS NOT ZERO *
GRIF 80 C     * APPLY THE ORIGINAL GRIFFITH IF THE INITIAL FLAW IS OPEN *
GRIF 81 C     *****
GRIF 82      PQADD=PP+Q
GRIF 83      PQSUB=PP-Q
GRIF 84      X=-0.5*PQSUB/PQADD
GRIF 85      SIGYY=0.5*(PQADD+PQSUB*X)
GRIF 86      IF(SIGYY.LT.0.) GO TO 200
GRIF 87      SIGN=-PQSUB**2/(0.*PQADD)
GRIF 88      DSIG=SIGN-SIGI
GRIF 89      IF((JCK.EQ.1).AND.(ABS(DSIG).LT.TOLL)) GO TO 150
GRIF 90      IF(DSIG.LT.TOLL) GO TO 600
GRIF 91      150 CONTINUE
GRIF 92      WX= ARCCOS(X)
GRIF 93      W=0.5*WX
GRIF 94      JNT(N)=2
GRIF 95      Z=G/PP
GRIF 96      S1= SIN(W)
GRIF 97      S2=S1**2
GRIF 98      SS= SIN(WX)
GRIF 99      ZZZ=Z*Z-1.0
GRIF 100     Z=Z-1.0

```

```

GRIF101      ARG=(Z*S2+1.0- SQRT(ZZ*S2+1.0))/(0.5*Z*SS)
GRIF102      EPSX= ATAN(ARG)-W
GRIF103      GO TO 300
GRIF104      200 CONTINUE
GRIF105 C     *****
GRIF106 C     * APPLY THE MODIFIED GRIFFITH IF THE INITIAL FLAW IS CLOSED
GRIF107 C     *****
GRIF108      IF(MD.EQ.0) GO TO 240
GRIF109 C     CHECK SHEAR STRENGTH
GRIF110      SIGN=0.5*(PQSUB*SQRT(1.+XN**2)+PQADD*XN)
GRIF111      DSIG=SIGN-SIGT
GRIF112      IF((JCK.EQ.1).AND.(ABS(DSIG).LT.TOLL)) GO TO 220
GRIF113      IF(DSIG.LT.TOLL) GO TO 400
GRIF114      220 SIGM=.5*PQADD
GRIF115      TAU=.5*PQSUB
GRIF116      IF(TAU.EQ.0.) GO TO 600
GRIF117      WX=ATAN(XN)
GRIF118      X=((SIGM+SIGT/XN)/TAU)*SIN(WX)
GRIF119      IF(X.GT.1.) X=1.
GRIF120      AX=ARSIN(X)
GRIF121      B1=-.5*(AX+WX)
GRIF122      B2=.5*(3.1416+AX-WX)
GRIF123      BW=.5*ATAN2(1.,XN)
GRIF124 C     CHECK BOUNDS OF ANGLE
GRIF125      IF((BW.GE.B1).AND.(BW.LE.B2)) GO TO 230
GRIF126      GO TO 600
GRIF127      230 XN=XNU(MTYPE)
GRIF128      SIGM=.25*(PQSUB*SQRT(1.+XN**2)+PQADD*XN)
GRIF129      IF(SIGN.GE.SIGM) GO TO 240
GRIF130      JNT(N)=5
GRIF131      GO TO 270
GRIF132      240 CONTINUE
GRIF133      X=1.0/XN
GRIF134      WX= ATAN(X)
GRIF135      SIGYY=0.5*(PQADD+PQSUB* COS(WX))
GRIF136      IF(SIGYY.GE.0.1) GO TO 600
GRIF137      W=0.5*WX
GRIF138      SIGN=0.25*(PQSUB* SQRT(1.+XN**2)+PQADD*XN)
GRIF139      DSIG=SIGN-SIGT
GRIF140      IF((JCK.EQ.1).AND.(ABS(DSIG).LT.TOLL)) GO TO 250
GRIF141      IF(DSIG.LT.TOLL) GO TO 600
GRIF142      250 CONTINUE
GRIF143      JNT(N)=3
GRIF144      270 CONTINUE
GRIF145      EPSX=0.7854-W
GRIF146      300 CONTINUE
GRIF147      MTAG(N)=1
GRIF148      EFTA(N)=R/57.296-EPSX
GRIF149      ANG=57.296*BETA(N)
GRIF150      IF(ANG.GT.90.0) BETA(N)=EFTA(N)-3.1416

```

```

GRIF151      IF(ANG.LT.-90.) BETA(N)=BETA(N)+3.1416
GRIF152      320 CONTINUE
GRIF153      IF(KCHECK.EQ.0) GO TO 600
GRIF154 C     *****
GRIF155 C     *   EVALUATE STRESS RATIO FOR EACH ELEMENT   *
GRIF156 C     *****
GRIF157      A0=SIG1(N,1)+SIG1(N,2)
GRIF158      B0=SIG1(N,1)-SIG1(N,2)
GRIF159      A1=SIG1(1)+SIG1(2)
GRIF160      B1=SIG1(1)-SIG1(2)
GRIF161      C0=B0**2+4.0*SIG1(N,3)**2
GRIF162      C1=B0*B1+4.0*SIG1(N,3)*SIG1(3)
GRIF163      C2=B1**2+4.0*SIG1(3)**2
GRIF164      IJNT=JNT(N)+1
GRIF165      GO TO (600,350,400,500,340,550),IJNT
GRIF166      340 C4=X**4
GRIF167      C1=C1*C4
GRIF168      C2=C2*C4
GRIF169      350 SIGT2=2.*SIG1
GRIF170      DSIGT=SIGT2-A0
GRIF171      AAA=A1**2-C2
GRIF172      BB=-A1*DSIGT-C1
GRIF173      CC=DSIGT**2-C0
GRIF174      GO TO 540
GRIF175      400 CONTINUE
GRIF176      AAA=C2
GRIF177      BB=C1+4.0*SIGT*A1
GRIF178      CC=8.0*SIGT*A0+CC
GRIF179      GO TO 550
GRIF180      500 CONTINUE
GRIF181      XN2=XN**2
GRIF182      RXN2=1.0+XN2
GRIF183      AAA=RXN2*C2-XN2*A1**2
GRIF184      IF(JNT(N).EQ.5) GO TO 520
GRIF185      BB=RXN2*C1-XN2*A1*A0+2.0*SIGT*A1*XN
GRIF186      CC=RXN2*C0-XN2*A0*A0+4.0*SIGT*A0*XN- 4.0*SIGT**2
GRIF187      GO TO 550
GRIF188      520 BB=RXN2*C1-XN2*A1*A0+4.0*SIGT*A1*XN
GRIF189      CC=RXN2*C0-XN2*A0*A0+8.0*SIGT*A0*XN-16.0*SIGT**2
GRIF190      550 CONTINUE
GRIF191      DD=BB**2-AAA*CC
GRIF192      IF(DD.GE.0.0) GO TO 560
GRIF193      GO TO 600
GRIF194      560 ROOT= SQRT(DD)
GRIF195      RI=(-BB+ROOT)/AAA
GRIF196      RJ=(-BB-ROOT)/AAA
GRIF197      RATIO(N)=RI
GRIF198      IF((RATIO(N).LT.TOLI ).AND.(RJ.GE.TOLI )) RATIO(N)=PJ
GRIF199      IF((RATIO(N).GT.1.).AND.(RJ.LT.1.)) RATIO(N)=RJ
GRIF200      IF( ABS(1.-RATIO(N)).LE.TOLI ) RATIO(N)=1.0

```

```

GRIF201      IF (RATIO(N).LT.TOL) RATIO(N)=TOL
GRIF202      IF (RATIO(N).GT.1.) RATIO(N)=1.
GRIF203      GOO CONTINUE
GRIF204      SIGNM(N)=SIGN
GRIF205      IF (MTAG(N).NE.MTAG1) JCHCK=1
GRIF206      IF (MTAG1.EQ.2) MTAG(N)=2
GRIF207      IF (MTAG(N).NE.1) RATIO(N)=1.0
GRIF208      RETURN
GRIF209 C
GRIF210      END

```

```

SCAL 1      SUBROUTINE SCALF(IX,MTAG,NTAG, JNT,RATIO,BETA,SIGI,SIGNM,EPSI,CU,
SCAL 2      . B,A)
SCAL 3 C
SCAL 4      COMMON/ONE/ NUMNP,NUMEL,NUMMAT,NUMPC,NPC,MBAND,NUMBLK,NL,MTYPE,N,
SCAL 5      . VOL,ACELK,ACELZ,Q,HED(18),STOP,SR,SR1,TOTAL,TOL,TOL1,XC,YC,
SCAL 6      . TEMP,SIGI,SIGDI,SIGDJ,
SCAL 7      . LLL,III,JJJ,JCK,KCHECK,JCHECK,NPLANE,NUMER,NCODE,JA,NFO
SCAL 8      . NBAND,NCRACK,NSTEP,N15,M7,NTOT
SCAL 9      COMMON/TWO/ C(3,3),S(10,10),SIG(6),P(8),ST(3,10),RR(5),ZZ(5),
SCAL 10     . LM(4),EE(3),EPS(3)
SCAL 11     COMMON/THREE/ E(8,4,8),EO(4,8),TENS(8),XNU(8),RD(8),EPST(8),
SCAL 12     . MTC(8),NTC(8),MID(8)
SCAL 13     DIMENSION IX(NUMEL,5),MTAG(NUMEL),NTAG(NUMEL),JNT(NUMEL),RATIO(NUM
SCAL 14     .EL),BETA(NUMEL),SIGI(NUMEL,6),SIGNM(NUMEL),FPSI(NUMEL,6),CU(NEQ)
SCAL 15     DIMENSION B(NL),A(NL,NBAND)
SCAL 16     DIMENSION UX(8),UY(8),F(4)
SCAL 17     KIK=0
SCAL 18     JCK=1
SCAL 19     ICHK=1
SCAL 20     ICHECK=0
SCAL 21     IF(III.GT.0) GO TO 350
SCAL 22     JCK=0
SCAL 23     KCHECK=1
SCAL 24     KKK=0
SCAL 25     SRR=1.
SCAL 26     SR1=1.0
SCAL 27     WRITE(6,2005) LLL
SCAL 28     GO TO 600
SCAL 29 100 CONTINUE
SCAL 30     NKT=0
SCAL 31     KKK=KKK+1
SCAL 32     SR=1.0
SCAL 33     DO 200 M=1,NUMEL
SCAL 34 200 IF(RATIO(M).LT.SR) SR=RATIO(M)
SCAL 35     SR3=SR
SCAL 36     TOLL=0.
SCAL 37     DO 210 M=1,NUMEL
SCAL 38     IF(MTAG(M).NE.1) GO TO 210
SCAL 39     MTYPE=IX(M,5)
SCAL 40     TOLJ=0.005
SCAL 41     TOLK=ABS(RATIO(M)-SR)
SCAL 42     IF(TOLK.GT.TOLJ) GO TO 210
SCAL 43     IF(TOLK.GT.TOLL) TOLL=TOLK
SCAL 44 210 CONTINUE
SCAL 45     IF(SR3.LT.1.) SR3=SR+TOLL
SCAL 46     IF(SR3.GT.1.) SR3=1.0
SCAL 47     DO 220 M=1,NUMEL
SCAL 48 220 IF( ABS(RATIO(M)-SR).LE.TOLJ ) RATIO(M)=SR3
SCAL 49     SR=SR3
SCAL 50     IF(SR3.NE.1.0) GO TO 240

```

```

SCAL 51      KCHECK=0
SCAL 52      SR=SR*SR1
SCAL 53      GO TO 250
SCAL 54 C
SCAL 55 C      INTERPOLATE OR EXTRAPOLATE SCALING FACTOR
SCAL 56 C
SCAL 57      240 CALL      INTER (IX,MTAG,JNT,RATIO,BETA,SIG1,SIGNM,EPSI,CU,E,A)
SCAL 58      250 SR1=SR
SCAL 59      DO 280 M=1,NUMPL
SCAL 60      IF (MTAG(M).GT.1) GO TO 280
SCAL 61      IF (MTAG(M).EQ.1) NKT=NKT+1
SCAL 62      IF (RATIO(M).EQ.SR3) GO TO 280
SCAL 63      IF (MTAG(M).EQ.1) NKT=NKT-1
SCAL 64      RATIO(M)=1.0
SCAL 65      BETA(M)=0.
SCAL 66      JNT(M)=0
SCAL 67      MTAG(M)=0
SCAL 68      280 CONTINUE
SCAL 69      IF (KCHECK.NE.0) GO TO 345
SCAL 70      KKK=KKK-1
SCAL 71      SR=SR*(1.-TOTAL)
SCAL 72      KIK=0
SCAL 73      IF ((KKK.EQ.0).AND.(SR.EQ.1.)) KIK=1
SCAL 74      WRITE(6,2006)
SCAL 75      WRITE(6,2002) LLL,SR,KKK
SCAL 76      GO TO 350
SCAL 77      345 WRITE(6,2003) LLL,KKK,SR
SCAL 78      IF (SR.GE.0.0001) GO TO 350
SCAL 79      WRITE(6,2004) SR
SCAL 80      SR=0.
SCAL 81      RETURN
SCAL 82      350 SR=SR1
SCAL 83      IF (111.GT.0) SR=1.0
SCAL 84      GO TO 600
SCAL 85      352 IF ((1CHK.EQ.0).AND.(KCHECK.EQ.1)) GO TO 100
SCAL 86 C      *****
SCAL 87 C      * SCALE DOWN DISPLACEMENTS *
SCAL 88 C      *****
SCAL 89      IF ((111.EQ.0).AND.(KCHECK.EQ.0)) GO TO 357
SCAL 90      GO TO 358
SCAL 91      357 TOTAL=TOTAL+SR
SCAL 92      IF ((TOTAL.GE..999).OR.(SR.EQ.0.0)) TOTAL=1.0
SCAL 93      358 CONTINUE
SCAL 94      IF ((111.GT.0).AND.(1CHECK.EQ.0)) GO TO 550
SCAL 95      IF (KIK.EQ.1) GO TO 359
SCAL 96      WRITE(6,2006)
SCAL 97      WRITE(6,2000)
SCAL 98      359 DO 500 M=1,NUMNP
SCAL 99      N1=M-1
SCAL 100      N2=N1+1

```

```

SCAL101      BX=SRR*B(N1)
SCAL102      BY=SRR*B(N2)
SCAL103      IF(ICHECK.EQ.1) GO TO 360
SCAL104      BXT=BX+CU(N1)
SCAL105      BYT=BY+CU(N2)
SCAL106      GO TO 370
SCAL107 360  CU(N1)=CU(N1)+BX
SCAL108      CU(N2)=CU(N2)+BY
SCAL109      BXT=CU(N1)
SCAL110      BYT=CU(N2)
SCAL111 370  IF(KIK.EQ.1) GO TO 500
SCAL112      WRITE(6,1000) M,BX,BY,BXT,BYT
SCAL113 500  CONTINUE
SCAL114      KIK=0
SCAL115 550  CONTINUE
SCAL116      ICHK=0
SCAL117      IF(III.NE.0) GO TO 900
SCAL118      IF(KCHECK.EQ.0) GO TO 900
SCAL119      GO TO 100
SCAL120 C *****
SCAL121 C * SCALE DOWN STRESSES *
SCAL122 C *****
SCAL123 600  CONTINUE
SCAL124      SM=0.
SCAL125      JCHK=0
SCAL126      KCK=0
SCAL127      IF((KKK.GT.0).AND.(KCHECK.EQ.1)) KCK=1
SCAL128      IF((KIK.EQ.1).OR.(KCK.EQ.1)) GO TO 605
SCAL129      WRITE(6,2001)
SCAL130 605  CONTINUE
SCAL131      DO 600 M=1,NUMEL
SCAL132      ID=M
SCAL133 C *****
SCAL134 C * READ ELEMNT INFORMATION FROM FILE 2 *
SCAL135 C *****
SCAL136      READ(2,ID) ((C(IJK,JJK),JJK=1,3),EE(IJK),IJK=1,3)
SCAL137      1,((S(JJI,KKI),KKI=1,8),JJI=1,8),((ST(IKK,JKK),JKK=1,8),IKK=1,3)
SCAL138      2,((RR(JII),ZZ(JII),JII=1,4),XC,YC,TEMP,VOL,MTYPE,N
SCAL139      DO 640 I=1,3
SCAL140 640  SIG(1)=SRR*S(01(M,1+3)+SIGI(M,1)
SCAL141      IF(IX(M,3).NE.IX(M,2)) GO TO 650
SCAL142      DX=RR(2)-RR(1)
SCAL143      DY=ZZ(2)-ZZ(1)
SCAL144      SIG(6)=90.
SCAL145      IF(DX.NE.0.) SIG(6)=57.296*ATAN2(DY,DX)
SCAL146      SIG(4)=SIG(1)
SCAL147      SIG(5)=0.
SCAL148      GO TO 765
SCAL149 650  CC=(SIG(1)+SIG(2))/2.0
SCAL150      BB=(SIG(1)-SIG(2))/2.0

```



```

SCAL151      CR= SQRT(BB**2+SIG(3)**2)
SCAL152      SIG(4)=CC+CR
SCAL153      SIG(5)=CC-CR
SCAL154      SIG(6)=0.0
SCAL155      IF((BB.EQ.0.).AND.(SIG(3).EQ.0.)) GO TO 660
SCAL156      SIG(6)=28.648* ATAN2(SIG(3),BB)
SCAL157  660 MTYPE=1X(M,5)
SCAL158      IF(MTC(MTYPE).EQ.0) GO TO 765
SCAL159      IF(111.EQ.0) GO TO 755
SCAL160      IF((MTAG(M).EQ.3).AND.(SIG(4).GT.  TENS(MTYPE))) MTAG(M)=2
SCAL161      IF(MTAG(M).EQ.2) JCHECK=1
SCAL162      IF(MTAG(M).EQ.2) GO TO 765
SCAL163      CALL      GRIFTH (MTAG,JNT,RATIO,BETA,SIG1,SIGNM)
SCAL164      GO TO 765
SCAL165  755 CONTINUE
SCAL166      IF(KCHECK.EQ.1) GO TO 757
SCAL167      IF((MTAG(M).EQ.3).AND.(SIG(4).GT.  TENS(MTYPE))) MTAG(M)=2
SCAL168      IF((MTAG(M).EQ.1).OR.(MTAG(M).EQ.2)) JCHECK=1
SCAL169      IF(MTAG(M).GT.2) GO TO 765
SCAL170  757 CALL      GRIFTH (MTAG,JNT,RATIO,BETA,SIG1,SIGNM)
SCAL171  765 SN=0.
SCAL172      IF(KCHECK.EQ.1) GO TO 770
SCAL173      IF((MTAG(M).EQ.0).OR.(MTAG(M).GT.2)) GO TO 770
SCAL174      SN=SIGNM(M)/FLNS(MTYPE)
SCAL175      IF(SIGNM(M).GT.SM) SM=SIGNM(M)
SCAL176  770 IF(KCK.EQ.1) GO TO 800
SCAL177      IF((SN.GE.1.).OR.(RATIO(M).GT.1.)) RATIO(M)=1.
SCAL178      IF(K1K.EQ.1) GO TO 800
SCAL179      ANG=57.296*BETA(M)
SCAL180      WRITE(6,1001) M,XC,YC,(SIG(I),I=1,6),MTAG(M),JNT(M),ANG,RATIO(M)
SCAL181      *,SN
SCAL182  800 CONTINUE
SCAL183      IF(111.NE.0) GO TO 830
SCAL184      IF(KCHECK.NE.0) GO TO 828
SCAL185      IF(JJJ.NE.0) GO TO 828
SCAL186 C
SCAL187 C *****
SCAL188 C * (CHECK CRACK CLOSURE WHEN 111=0,KCHECK=0 & JJJ=0 .
SCAL189 C *****
SCAL190 C
SCAL191      WRITE(6,2007)
SCAL192      DO 827 N=1,NUMEL
SCAL193      IF(MTAG(N).LT.3) GO TO 827
SCAL194      IF(1X(N,2).EQ.1X(N,3)) GO TO 827
SCAL195      ID=N
SCAL196      MM=4
SCAL197      IF(1X(N,1).EQ.1X(N,4)) MM=3
SCAL198      FIND(2*ID)
SCAL199      KTAG=0
SCAL200      DO 802 I=1,MM

```

```

SCAL201      UX(I)=0.
SCAL202      802 UY(I)=0.
SCAL203      READ(2,10) ((C(IJK,JJK),JJK=1,3),EE(IJK),IJK=1,3)
SCAL204      1,((S(JJI,KK1),KK1=1,8),JJI=1,8),((SI(IKK,JKK),JKK=1,8),IKK=1,3)
SCAL205      2,((RR(JII),ZZ(JII),JII=1,4),XC,YC,TEMP,VOL,MTYPE,N
SCAL206      ICK=0
SCAL207      EPT=FPST(MTYPE)
SCAL208 C
SCAL209 C      DECOMPOSE JOINT OPENING INTO X AND Y COMPONENTS
SCAL210 C
SCAL211      EPSX=1.5708+BFFA(N)
SCAL212      IF(EPSX.GE.1.5708) EPSX=1.56
SCAL213      804 SLOPE=TAN(EPSX)
SCAL214      DO 806 I=1,MM
SCAL215      806 F(I)=SLOPE*(RR(I)-XC)-(ZZ(I)-YC)
SCAL216      EPSX=BETA(N)
SCAL217      IF(BETA(N).LT.0.) EPSX=3.1416+EPSX
SCAL218      IF(KTAG.EQ.1) EPSX=EPSX-1.5708
SCAL219      DUX=EPT*COS(EPSX)
SCAL220      DUY=EPT*SIN(EPSX)
SCAL221      DO 808 I=1,MM
SCAL222      IF(F(I).LT.0.) GO TO 808
SCAL223      UX(I)=UX(I)+DUX
SCAL224      UY(I)=UY(I)+DUY
SCAL225      808 CONTINUE
SCAL226      IF(KTAG.EQ.1) GO TO 810
SCAL227      IF(MTAG(N).EQ.3) GO TO 812
SCAL228      KTAG=1
SCAL229      EPSX=BETA(N)
SCAL230      GO TO 804
SCAL231      810 EPSX=EPSX+1.5708
SCAL232      812 CONTINUE
SCAL233 C
SCAL234 C      EVALUATE EQUIVALENT STRAINS
SCAL235 C
SCAL236      DO 816 I=1,3
SCAL237      RR(I)=0.0
SCAL238      DO 816 J=1,MM
SCAL239      K=2+J
SCAL240      816 RR(I)=RR(I)+SI(I,K)*UY(I)+ST(I,K-1)*UX(I)
SCAL241      817 CONTINUE
SCAL242      C1=COS(EPSX)
SCAL243      S1=SIN(EPSX)
SCAL244      SC=C1*S1
SCAL245      C1=C1*C1
SCAL246      S1=S1*S1
SCAL247      IF(ICK.EQ.1) GO TO 818
SCAL248      DP=SC*RR(3)
SCAL249      EPT1=ABS(C1*RR(1)+S1*RR(2)+DP)
SCAL250      EPT2=ABS(S1*RR(1)+C1*RR(2)-DP)

```

```

SCAL251      IF(BETA(N).GT.0.) GO TO 816
SCAL252      EPSX=BETA(N)
SCAL253      ICK=1
SCAL254      GO TO 817
SCAL255 C
SCAL256 C      EVALUATE ELEMENT PRINCIPAL STRAINS, CHECK CLOSURE POSSIBILITY
SCAL257 C
SCAL258      818 EPS(1)=SRR*EPS1(N,4)+EPS1(N,1)
SCAL259      EPS(2)=SRK*EPS1(N,5)+EPS1(N,2)
SCAL260      EPS(3)=SPR*EPS1(N,6)+EPS1(N,3)
SCAL261      DP=SC*EPS(3)
SCAL262      EPS1=C1*EPS(1)+S1*EPS(2)+DP
SCAL263      EPS2=S1*EPS(1)+C1*EPS(2)-DP
SCAL264      C1=EPT1+EPS1
SCAL265      S1=EPT2+EPS2
SCAL266      IF(C1.GT.0.) GO TO 824
SCAL267      IF(S1.GT.0.) GO TO 823
SCAL268      MTAG(N)=0
SCAL269      GO TO 816
SCAL270      823 MTAG(N)=MTAG(N)-3
SCAL271      IF(MTAG(N).EQ.0) GO TO 826
SCAL272      BETA(N)=BETA(N)+1.5708
SCAL273      IF(BETA(N).GT.1.5708) BETA(N)=BETA(N)-3.1416
SCAL274      GO TO 826
SCAL275      824 IF(S1.GT.0.) GO TO 827
SCAL276      IF(MTAG(N).EQ.3) GO TO 827
SCAL277      MTAG(N)=1
SCAL278      826 JCHECK=1
SCAL279      NTAG(N)=1
SCAL280      ANG=57.296*BETA(N)
SCAL281      WRITE(6,2015)
SCAL282      WRITE(6,2020) N,MTAG(N),ANG,EPT,EPT1,EPT2,EPS1,EPS2,C1,S1
SCAL283      827 CONTINUE
SCAL284      828 IF((OTAL..0.1.) ICHECK=1
SCAL285      IF((KCHECK.EC.0).AND.(NKT.LE.0)) ICHECK=1
SCAL286      GO TO 840
SCAL287 C
SCAL288 C      SEARCH FOR MAX. SIGNM
SCAL289 C
SCAL290      830 DO 835 N=1,NUMEL
SCAL291      IF( MTAG(N).GT.2) GO TO 835
SCAL292      DN=ABS(SIGNM(N)-SM)
SCAL293      IF(DN.LE.TOL) GO TO 835
SCAL294      IF(MTAG(N).NE.2) GO TO 834
SCAL295      MTAG(N)=3
SCAL296      DA=1.5708
SCAL297      IF(BETA(N).GT.0.) DA=-DA
SCAL298      BETA(N)=BETA(N)+DA
SCAL299      GO TO 835
SCAL300      834 MTAG(N)=0

```

```

SCAL301      JNT(N)=0
SCAL302      BETA(N)=0.
SCAL303      RATIO(N)=1.
SCAL304      835 ICHECK=1
SCAL305      840 IF(ICHECK.EQ.1) ICHECK=0
SCAL306      IF(ICHECK.EQ.1) GO TO 850
SCAL307      GO TO 352
SCAL308      850 DO 860 M=1,NUMFL
SCAL309      DO 860 I=1,3
SCAL310      SIGI(M,I)=SRR*SIGI(M,I+3)+SIGI(M,I)
SCAL311      EPSI(M,I)=SRR*EPSI(M,I+3)+EPSI(M,I)
SCAL312      860 CONTINUE
SCAL313      GO TO 352
SCAL314      900 CONTINUE
SCAL315      WRITE(6,2010) TOTAL
SCAL316      RETURN
SCAL317      1000 FORMAT(112,4E20.7)
SCAL318      1001 FORMAT(17,2F8.2,1P5E12.4,0P1F7.2,215,F7.2,1X,F7.4,F7.2)
SCAL319      2000 FORMAT(12HON.P.NUMBER ,17X,3HDUX,17X,3HDUY,18X,2HUX,18X,2HUY)
SCAL320      2001 FORMAT (7HDEL.ND. 7X 1HX 7X 1HY 4X 8HX-STRESS 4X 8HY-STRESS 3X
SCAL321      1 9HXY-STRESS,2X,10HMAX-STRESS,2X,10HMIN-STRESS,7H ANGLE
SCAL322      2,1X,4HMTAG,2X,3HJNT,3X,4HBETA,2X,5HRATIO,2X,5HSIGNM)
SCAL323      2002 FORMAT(/ ' LOAD INCREMENT NO.',15, / ' LOAD APPLIED AS A FRACTION ()
SCAL324      *F TOTAL LOAD =' ,F9.5, / ' NUMBER OF BETA STABILITY ITERATIONS=' ,15 /)
SCAL325      2003 FORMAT(1H0, ' LOAD INCREMENT NO.',15, ' STABILITY ITERATION NO.',
SCAL326      * 13, ' SCALING FACTOR =' ,G10.4)
SCAL327      2004 FORMAT(/ ' SR=' ,G10.4, ' CALL EXIT')
SCAL328      2005 FORMAT(1H1, ' INITIAL STATE FOR LOAD INCREMENT',15)
SCAL329      2006 FORMAT(1H1)
SCAL330      2007 FORMAT(/ ' CHECK CRACK OPENING STRAINS ')
SCAL331      2010 FORMAT(1H0, ' LOAD ACCUMULATED AS A FRACTION OF THE TOTAL IS',G15.5
SCAL332      *//)
SCAL333      2015 FORMAT(1H0, ' N TAG',8X,'BETA',8X,' EPT',8X,'EPT1',8X,'EPT2',8X,
SCAL334      *'EPS1',8X,'EPS2',7X,'DEPS1',7X,'DEPS2')
SCAL335      2020 FORMAT(13,15,F12.2,6E12.4)
SCAL336      2025 FORMAT(18,2F10.4,3E12.5)
SCAL337      END

```

```

INTR 1      SUBROUTINE INTER (IX,MTAG,JNT,RATIO,BETA,SIGI,SIGNM)
INTR 2 C
INTR 3      COMMON/ONE/ NUMNP,NUMEL,NUMHAT,NUMPC,NPC,MBAND,NUMBLK,NL,MTYPE,N,
INTR 4      . VCL,ACELR,ACELZ,Q,HEO(18),STOP,SR,SR1,TOTAL,YUL,TOLI,XC,YC,
INTR 5      . TEMP,SIGN,SIGDI,SIGDJ,
INTR 6      . LLL,III,JJJ,JCK,KCHECK,JCHECK,NPLANE,NUMER,NCODE,JA,NEO
INTR 7      . NBAND,NCRACK,NSTEP,N15,M7,NTOT
INTR 8      COMMON/TWO/ C(3,3),S(10,10),SIG(6),P(8),ST(3,10),RR(5),ZZ(5),
INTR 9      . LM(4),LE(3),EPS(3)
INTR 10     COMMON/THREE/ E(8,4,8),EO(4,8),TENS(8),XNU(8),RO(8),EPST(8),
INTR 11     . MTC(8),NIC(8),MID(8)
INTR 12     DIMENSION IX(NUMEL,5),MTAG(NUMEL),JNT(NUMEL),RATIO(NUMEL),
INTR 13     . BETA(NUMEL),SIGI(NUMEL,6),SIGNM(NUMEL)
INTR 14     JCK=1
INTR 15     SRR=SR1
INTR 16     IJJ=0
INTR 17     DO 200 N=1,NUMEL
INTR 18     IF(RATIO(N).NE.SR) GO TO 200
INTR 19     SR3=RATIO(N)
INTR 20     ELTA=BETA(N)
INTR 21     MTAGI=MTAG(N)
INTR 22     JNTI=JNT(N)
INTR 23     SR=SR*SR1
INTR 24     MTYPE=IX(N,5)
INTR 25     SIGT=TENS(MTYPE)
INTR 26     IF(IX(N,2).EQ.IX(N,3)) GO TO 200
INTR 27     50 CONTINUE
INTR 28     DO 100 I=1,3
INTR 29     100 SIG(I)=SR3*SIGI(N,I+3)+SIGI(N,I)
INTR 30     CC=(SIG(1)+SIG(2))/2.0
INTR 31     EB=(SIG(1)-SIG(2))/2.0
INTR 32     CR= SQRT(EB**2+SIG(3)**2)
INTR 33     SIG(4)=CC+CR
INTR 34     SIG(5)=CC-CR
INTR 35     SIG(6)=0.0
INTR 36     IF((EB.EQ.0.).AND.(SIG(2).EQ.0.)) GO TO 150
INTR 37     SIG(6)=20.648* ATAN2(SIG(3),EB)
INTR 38     150 CONTINUE
INTR 39     CALL GRIFTH (MTAGI,JNTI,RATIO,BETA,SIGI,SIGNM)
INTR 40     IF(IJJ.GI.0) GO TO 300
INTR 41     SIGDI=SIGN-SIGT
INTR 42     SRR=SR
INTR 43     IF(SRR.LT.TOLI) SPR=TOLI
INTR 44     IJJ=1
INTR 45     GO TO 50
INTR 46     200 CONTINUE
INTR 47     300 CONTINUE
INTR 48     IF (IX(N,2).EQ.IX(N,3)) GO TO 400
INTR 49     RATIO(N)=SR3
INTR 50     ELTA(N)=BETA I

```

```

INTR 51      MTAG(N)=MTAGI
INTR 52      JN1(N)=JNT1
INTR 53      SIGDJ=SIGN-SIGT
INTR 54      IF(( ABS(SIGDI).LE.TOL ).OR.( ABS(SIGDJ).LE.TOL )) GO TO 350
INTR 55      SR=(SR1*SIGDJ-SR*SICDI)/(SIGDJ-SIGDI)
INTR 56      GO TO 400
INTR 57      350 CONTINUE
INTR 58      IF( ABS(SIGDI).GT.TOL ) GO TO 370
INTR 59      SR=SK1
INTR 60      GO TO 390
INTR 61      370 IF( ABS(SIGDJ).GT.TOL ) GO TO 400
INTR 62      SR=SRR
INTR 63      390 CONTINUE
INTR 64      400 CONTINUE
INTR 65      IF(SR.GE.0.99 ) SR=1.0
INTR 66      IF(SR.LT.0.001) SR=0.001
INTR 67      IF(SR.EQ.1.0) KCHECK=0
INTR 68      TOLJ=0.01
INTR 69      DA=SR-SRR
INTR 70      DB=SR-SR1
INTR 71      IF((ABS(DA).LE.TOLJ).OR.(ABS(DB).LE.TOLJ)) KCHECK=0
INTR 72      WRITE(6,1000, SR1,SRR,SR
INTR 73      RETURN
INTR 74      1000 FORMAT(// ' OLD SR=',G14.4, ' NEW SR=',G14.4, 10X, ' INTERPOLATED SR=
INTR 75      *,G14.4//)
INTR 76      C
INTR 77      END

```

```

MOD1 1      SUBROUTINE MODIFY(A,B,NEQ,MEAND,NBAND,N,U)
MOD1 2 C
MOD1 3 C
MOD1 4      DIMENSION B(N,0),A(NEQ,NBAND)
MOD1 5      DO 250 M=1,MEAND
MOD1 6      K=N-M+1
MOD1 7      IF(K) 235,235,230
MOD1 8 230   B(K)=B(K)-A(K,M)*U
MOD1 9      A(K,M)=0.0
MOD1 10 235  K=N+M-1
MOD1 11      IF(NEQ-K) 250,240,240
MOD1 12 240   B(K)=B(K)-A(N,M)*U
MOD1 13      A(N,M)=0.0
MOD1 14 250  CONTINUE
MOD1 15      A(N,1)=1.0
MOD1 16      B(N)=U
MOD1 17      RETURN
MOD1 18      END

```



```

BANS 1      SUBROUTINE BANSOL (F,A,MM,NBANI,NUMBLK,NLI,JA)
BANS 2 C
BANS 3 C
BANS 4      DIMENSION B(NLI),A(NLI,NBAND)
BANS 5      NN=NLI/2
BANS 6      NL=NN+1
BANS 7      NH=N/2*NN
BANS 8      NB=0
BANS 9      NBK=1
BANS 10     FIND(1*1)
BANS 11     GO TO 150
BANS 12     100 NB=NB+1
BANS 13     DO 125 N=1,NN
BANS 14     NM=NN+N
BANS 15     B(N)=B(NM)
BANS 16     B(NM)=0.0
BANS 17     DO 125 M=1,MM
BANS 18     A(N,M)=A(NM,M)
BANS 19     125 A(NM,M)=0.0
BANS 20     IF (NUMBLK-NB) 150,200,150
BANS 21     150 READ (1*NBK) (B(N),(A(N,M),M=1,MM),N=NL,NH)
BANS 22     NNK=NBK
BANS 23     NBK=NBK+JA
BANS 24     IF (NB) 200,100,200
BANS 25     200 DO 300 N=1,NN
BANS 26     IF (A(N,1)) 225,300,225
BANS 27     225 B(N)=B(N)/A(N,1)
BANS 28     DO 275 L=2,MM
BANS 29     IF (A(N,L)) 230,275,230
BANS 30     230 C=A(N,L)/A(N,1)
BANS 31     I=N+L-1
BANS 32     J=0
BANS 33     DO 250 K=L,MM
BANS 34     J=J+1
BANS 35     250 A(I,J)=A(I,J)-C*A(N,K)
BANS 36     B(I)=B(I)-A(N,L)*B(N)
BANS 37     A(N,L)=C
BANS 38     275 CONTINUE
BANS 39     300 CONTINUE
BANS 40     NFK=NNK-JA
BANS 41     IF (NUMBLK-FC-NB) GO TO 410
BANS 42     WRITE (1*NBK) (B(N),(A(N,M),M=2,MM),N=1,NN)
BANS 43     NBK=NNK+JA
BANS 44     GO TO 100
BANS 45     410 DO 450 M=1,NN
BANS 46     N=NN+1-M
BANS 47     DO 425 K=2,MM
BANS 48     L=N+K-1
BANS 49     425 B(N)=B(N)-A(N,K)*B(L)
BANS 50     NM=N+NN

```

```

BANS 51      B(NM)=B(N)
BANS 52      450 A(NM,NB)=F(N)
BANS 53      NB=NB-1
BANS 54      IF(NB.LU.O) GO TO 500
BANS 55      READ (1,NBK) (B(N),(A(N,M),M=2,MM),N=1,NN)
BANS 56      NBK=NBK-JA
BANS 57      GO TO 410
BANS 58      500 K=0
BANS 59      DO 600 NB=1,NUMBLK
BANS 60      DO 600 N=1,NN
BANS 61      NM=N+NN
BANS 62      K=K+1
BANS 63      600 B(K)=A(NM,NB)
BANS 64      RETURN
BANS 65 C
BANS 66      END

```